Developments in the Built Environment 20 (2024) 100549

Contents lists available at ScienceDirect

Developments in the Built Environment

journal homepage: www.sciencedirect.com/journal/developments-in-the-built-environment

Automatic bridge inspection database construction through hybrid information extraction and large language models

Chenhong Zhang [a], Xiaoming Lei [b], Ye Xia [a,c,*], Limin Sun [a,c]

[a] Department of Bridge Engineering, Tongji University, Shanghai, China
[b] Department of Civil and Environmental Engineering, The Hong Kong Polytechnic University, Hong Kong China
[c] Shanghai Qi Zhi Institute, Shanghai, China

ARTICLE INFO

Keywords:
Bridge inspection data
Natural language processing
Information extraction
Large languge model
Pseudo label

ABSTRACT

Regular bridge inspections generate extensive reports that, while critical for maintenance, often remain underutilized due to their unstructured format. Traditional information extraction methods depend on intricate labeling systems that commonly require time-consuming and labor-intensive labeling. This paper presents a novel bridge inspection database construction method leveraging LLM-assisted information extraction. First, we introduce the pseudo-labelling method using a closed-source LLM to generate high-quality data. Then we propose the hybrid extraction pipeline to extract relevant information segments and process them by a generation-based IE model, fine-tuned on pseudo-labeled data. Finally, the extracted data is used to construct the bridge inspection database. The proposed method, validated with real-world data, not only demonstrates higher extraction precision than the closed-source LLM used for pseudo-labeling but also outperforms traditional methods in both data preparation time and extraction accuracy. This approach provides a scalable solution for more proactive and data-driven bridge maintenance strategies.

1. Introduction

Regular inspections serve as the fundamental method for assessing the structural conditions of bridges. This practice, widely implemented by bridge maintenance facilities around the globe, has led to the accumulation of inspection reports over the years. These reports documented the deterioration of numerous bridges over extended periods, offering a rich data repository invaluable for understanding deterioration patterns (Xia Lei et al., 2022; Lei Sun et al., 2024) and for formulating strategic maintenance plans (Lei Dong et al., 2023; Lai Dong et al., 2024).

While some developed countries have implemented sophisticated bridge management systems, such as the PONTIS(Thompson Small et al., 1998) system in the USA, to efficiently store and utilize historical inspection data, many developing countries still manage this data non-digitally, relying on paper-based inspection reports. This approach often results in reports being read only once before being discarded, leading to a significant waste of valuable information. The main difference between a structured inspection database and paper-based inspection reports lies in their accessibility to computers. Inspection reports, typically written in natural language, are not readily interpretable by computers (Li and Harris, 2019). Therefore, converting unstructured data from inspection reports into structured formats can significantly enhance the development of an organized bridge management system and improve maintenance planning (Lin Hu et al., 2016). For example, Feng et al.(Feng Wang et al., 2023) developed a natural language processing (NLP)-based machine learning approach to automate and accelerate the process of bridge condition rating, using data from inspection reports to predict bridge conditions with high accuracy and efficiency suitable for large-scale applications. Li et al.(Li Alipour et al., 2021) developed a data-driven framework using RNN encoders with attention mechanisms to automate bridge condition ratings and perform real-time quality control based on narrative descriptions from inspection reports, enhancing the consistency and accuracy of bridge management strategies. Liu et al.(Liu and El-Gohary, 2020) proposed a hybrid data fusion method that combines unsupervised named entity normalization and entropy-based numerical data fusion to effectively handle the dimensionality and sparsity issues in textual bridge inspection reports, significantly improving the accuracy of data-driven bridge condition predictions.

Extracting structured data from unstructured texts falls under the domain of Information Extraction (IE). IE techniques are designed to identify and extract critical information such as entities, relationships,

* Corresponding author. Department of Bridge Engineering, Tongji University, Shanghai, China.
E-mail addresses: chzhangacd@gmail.com, 2232508@tongji.edu.cn (C. Zhang), yxia@tongji.edu.cn (Y. Xia).

https://doi.org/10.1016/j.dibe.2024.100549
Received 19 April 2024; Received in revised form 28 August 2024; Accepted 28 September 2024
Available online 1 October 2024

or events within texts(Grishman, 2015). Given that IE encompasses many fundamental tasks in natural language processing, it has been extensively researched by scholars over the past decades. Rule-based IE methods, known for their simplicity and efficiency, were among the first to be developed and continue to be effective for specific tasks(Zhang and El-Gohary, 2016; Wu Lin et al., 2022). However, with the rapid advancements in deep learning, intelligent IE approaches have begun to outperform in various complex IE challenges. The widespread application of intelligent IE methods in tasks such as data extraction from inspection reports is hindered by the requirement for large volumes of training data. Open-source databases often lack coverage in specialized fields like civil engineering, let alone in niche areas such as bridge inspection data. To bridge this gap, some researchers have developed domain-specific corpora. For example, Zheng et al.(Zheng Lu et al., 2022) developed ARCBERT, a domain-specific language model for the architecture, engineering, and construction (AEC) sector, demonstrating how domain-specific pretraining improves performance on natural language processing tasks like text classification and named entity recognition within the AEC domain. However, creating a high-quality dataset can be prohibitively expensive for most researchers or bridge management personnel, thus limiting their ability to develop effective IE models.

Semi-supervised learning techniques present a solution to the challenge of limited data. Among these, pseudo-labeling is a popular method that leverages a small amount of labeled data alongside a larger volume of unlabeled data. This approach involves using a model trained on the available real data to generate pseudo-labels for the unlabeled data, thereby expanding the training dataset and enabling the development of more robust models. LLMs have recently been applied to generate pseudo-labeled data, showing promising results in general tasks like named entity recognition and text classification(Malik Bernard et al., 2024). However, the effectiveness of LLMs in generating pseudo-labeled data for industry-specific domains, for example the extraction of bridge inspection data, remains largely unexplored.

This paper introduces a method leveraging LLMs for the automatic construction of a bridge inspection database. Initially, we employ Gemini, a high-performance but closed-source LLM, to generate pseudo-labels for downstream training. Subsequently, we establish a hybrid information extraction pipeline specifically tailored for bridge inspection reports. This begins with a rule-based approach to identify and extract relevant segments containing key information from the inspection documents. Following this, we develop a generation-based IE model using an open-source LLM, GLM-4, to transform the extracted segments into structured data. Finally, we utilize the structured data to develop and construct a comprehensive bridge inspection database.

The primary innovation of this study includes following 3 aspects: 1) This study introduces a novel information extraction method that addresses the challenge of obtaining high-quality labels in the field of bridge maintenance. Traditional IE methods rely on complex labeling schemes, which often require time-consuming and costly manual labeling. Additionally, conventional BIO (Beginning, Inside, Outside) and entity-relation-triplet tagging schemes struggle to handle bridge maintenance data, especially when multiple, overlapping entities are involved. The proposed method, utilizing a generation-based IE model fine-tuned on pseudo-labels in an end-to-end manner, achieves high precision in defect data extraction without the need for extensive and complex manually labeled data. 2) While LLMs have been applied to general information extraction tasks, their use in domain-specific IE tasks like bridge maintenance is often costly and imprecise. Our proposed method fine-tunes a small, open-source LLM on pseudo-labeled data, achieving higher precision than a sophisticated closed-source LLM. Additionally, we identified the optimal settings for integrating the pseudo-labeling process with the fine-tuning of the generation-based IE model. 3) Although LLM-assisted pseudo-labeling processes have been validated in general domains, their applicability to bridge maintenance data had not been explored. This study determined the optimal pseudo-labels generation method from different prompting strategies and investigated the optimal prompting structures for bridge maintenance. These innovations set a new precedent for the effective use of LLMs in bridge maintenance, facilitating the transition to a more automated and intelligent bridge management system.

2. Related works

2.1. Semi-supervised learning and pseudo-labelling

Semi-supervised learning is a machine learning approach that combines a small amount of labeled data with a large amount of unlabeled data to enhance model training. Pseudo-labeling (PL) is a specific technique in semi-supervised learning where the predictions on unlabeled data are used as temporary labels to further refine its training. Various studies have proven PL successful across different domains. In the area of NLP, Sazzed et al. (2021) developed a cross-lingual hybrid methodology for sentiment analysis in Bengali, utilizing machine translation and pseudo-labels to train a classifier, significantly enhancing sentiment classification performance in this low-resource language. Zhang et al. (Zhang Li et al., 2023a) developed an end-to-end joint entity recognition and relation extraction system that utilizes a pseudo-graph structure, label reuse, and gating mechanisms to enhance accuracy and efficiency. Zhang et al.(Zhang Li et al., 2023b) developed the Task Relation Distillation and Prototypical pseudo label (RDP) method for Incremental Named Entity Recognition (INER) that effectively addresses catastrophic forgetting and background shift by implementing inter-task relation distillation and prototypical pseudo labeling.

Specifically, pre-trained models, with their inherent high-level knowledge, are extremely useful for generating pseudo-labeled data. For example, Huizinga et al. (Huizinga Kruithof et al., 2023) introduced CLaP, a semi-supervised learning approach that efficiently selects initial labels via clustering, utilizes consistent samples for pseudo-labeling, and applies a mixed loss training technique, significantly reducing human annotator burden. Kim et al. (Kim and Kang, 2022) proposed a text embedding augmentation method using adversarial training to prevent overfitting in fine-tuning large pre-trained language models for task-specific datasets, demonstrating effectiveness through benchmarks on text classification tasks.

LLMs, with their superior capabilities in understanding and generating natural language, surpass traditional language models and human annotators(Gilardi Alizadeh et al., 2023), making them ideal for annotating natural language texts. For example, Wen et al.(Wen Chen et al., 2021) developed a medical named entity recognition approach by combining a medical entity dictionary with domain-specific pre-trained language models and a pseudo labeling mechanism, achieving high F1 scores on Chinese electronic medical records. Malik et al.(Malik Bernard et al., 2024)]d eveloped a semi-supervised multi-label emotion classification model for French tweets using pseudo-labels generated by Chat-GPT, significantly enhancing accuracy by integrating these labels with manual annotations and demonstrating superior performance on a new dataset related to an urban industrial incident. Chien et al.(Chien and Chen, 2023)]introduced a collaborative pseudo labeling technique for prompt-based learning in few-shot classification tasks, implementing a teacher-student model that effectively mitigates model bias from incorrect pseudo labels, showing improved performance in sentiment classification and natural language inference.

2.2. Information extraction

Common IE challenges include tasks such as named entity recognition (NER), relation extraction (RE), event extraction (EE), and so on. They are widely utilized in the field of civil engineering, particularly in areas such as Building Information Modeling (BIM), and the maintenance and management of infrastructure. For example, Zheng et al.

(Zheng Zhou et al., 2024) proposed a novel approach to automatically evaluate and enhance the machine interpretability of building codes using a domain-specific language model and transfer learning, demonstrating significant improvements in text classification accuracy and interpretability assessments of regulatory documents. Zhang et al. (Zhang Chan et al., 2022) conducted a comprehensive systematic-bibliometric analysis on the integration of BIM and AI in the architecture-engineering-construction/facility management (AEC/FM) industry, identifying key trends, techniques, and future directions from 183 scholarly works. Wang et al.(Wang Issa et al., 2022) developed a NLP-based query-answering system for BIM information extraction, providing a virtual assistant architecture that supports construction project team members with an 81.9% accuracy on BIM-related queries. Yin et al.(Yin Tang et al., 2024) introduced a natural language understanding-based method for automating the ontological knowledge modeling of project-specific property concepts in BIM, significantly enhancing the alignment and integration of new property concepts with the Industry Foundation Classes (IFC) ontology. Zheng et al.(Zheng Zhou et al., 2022) developed a knowledge-informed framework for automated rule checking in the construction industry, utilizing NLP to establish an ontology, perform semantic alignment and conflict resolution, and generate SPARQL-based queries for model checking, significantly enhancing the accuracy and speed of rule interpretation compared to traditional methods.

With the rapid advancements in deep learning, modern research predominantly employs intelligent methods, which can be categorized to classification-based and generation based. For traditional classification-based intelligent IE approaches, two prevalent modeling strategies are used: sequence labeling and span classification.

Sequence labeling involves assigning a category label to each element in a sequence. For example, Huang et al.(Huang Xu et al., 2015) proposed the BI-LSTM-CRF model for sequence tagging in NLP, achieving near state-of-the-art results on POS, chunking, and NER datasets by effectively leveraging both past and future input features along with sentence-level tag information. Zheng et al. (2017) proposed a novel tagging scheme and end-to-end model that transform the joint extraction of entities and relations into a simple tagging problem, achieving superior results on a public dataset compared to traditional methods.

While sequence labeling assigns a label to each token individually, span classification assigns labels to sequences or ranges of tokens as whole units, making it favorable in more complex conditions. For span classification methods, Jiang et al. (2020) developed a unified model for natural language analysis by representing a wide array of tasks as span and relation annotations, effectively generalize across 10 diverse tasks with performance comparable to specialized models. Yu et al. (2020) redefined NER by applying a dependency parsing approach, using a biaffine model to predict both nested and flat entities, achieving state-of-the-art results on eight different NER corpora. Yan et al.(Yan Sun et al., 2023) introduced a Unified Token-pair Classification architecture for Information Extraction (UTC-IE), which simplifies and unifies various information extraction tasks into token-pair classifications, leveraging a novel transformer structure to model local interactions and improve task performance across multiple datasets.

Although classification-based methods in IE yield satisfactory performance, generation-based models offer greater flexibility, as they are typically not confined to specific tasks. One of the most established generation-based IE methods is Machine Reading Comprehension (MRC). MRC utilizes algorithms to comprehend and analyze textual content, extracting pertinent information or responding to specific queries posed by users. Levy et al. (2017) first demonstrated that relation extraction can be effectively approached through reading comprehension by associating each relation slot with natural-language questions, utilizing neural reading comprehension techniques for model learning, leveraging large datasets through crowd-sourcing and distant supervision, and enabling zero-shot learning for new, unseen relation types at test time. Then, Li et al. (2020) proposed a unified framework for both flat and nested NER by reformulating it as an MRC task, demonstrating significant performance improvements on various datasets through this novel approach.

Besides MRC approaches, other generation-based methods can also achieve satisfying performance on IE tasks leveraging pre-trained language models. For example, Yan et al. (2021) proposed a unified sequence-to-sequence framework to tackle flat, nested, and discontinuous NER subtasks simultaneously, leveraging pre-trained LLMs and a novel method for linearizing entities into sequences, achieving state-of-the-art or near state-of-the-art performance on eight English NER datasets. Hsu et al. (2022) developed DEGREE, a data-efficient model for low-resource event extraction that formulates the task as a conditional generation problem, utilizing manually designed prompts to guide the generation of summaries from which event information is deterministically extracted, demonstrating strong performance with minimal training data.

3. Methodology

In this paper, we present an automatic bridge inspection construction method using a novel hybrid information extraction pipeline to process unstructured textual data from inspection reports. First, we developed a rule-based extraction program to automatically identify and extract key information segments, consisting mainly of unstructured textual data, from the reports. We then used a closed-source large language model, Gemini, to generate pseudo-labels for the textual data, which helped training a generation-based information extraction model. Finally, the structured data generated by the model was used to build the bridge inspection database. The overall framework of the proposed method is shown in Fig. 1.

3.1. Hybrid extraction pipeline of inspection reports

Bridge inspection reports are typically lengthy documents that include various sections detailing the inspection procedures and outcomes for multiple bridges, with a single report potentially spanning hundreds of pages. Relying solely on machine learning models for the extraction process from such extensive documents is impractical. Additionally, given that these reports are often authored by the same agency and therefore follow similar formats, rule-based extraction methods can be particularly effective in these instances.

Consequently, we designed a hybrid extraction pipeline for extraction of bridge inspection reports. Initially, a rule-based method is employed to identify and extract target information, effectively segmenting the document into manageable portions. Subsequently, a generation-based IE model is applied to these segments to extract essential information. This two-step pipeline, as depicted in Fig. 2, combines the efficiency of rule-based methods with the flexibility of intelligent models, ensuring rapid and accurate information extraction from real-life inspection reports.

3.1.1. Rule-based extraction method

This section outlines a rule-based IE method tailored to identify and extract crucial information from bridge inspection reports. Considering the diversity of report formats among various inspection facilities, it's unfeasible to devise a universal set of extraction rules applicable to all reports. Nonetheless, experience indicates that these report formats usually contain common features, which can be leveraged to develop foundational extraction rules. Commonly, information regarding bridge defects, condition assessments, and structural attributes is organized in tables within the reports. For instance, Table 1 provides an example of how defect data is typically structured in these inspection documents.

For reports produced by the same inspection facility, the format of tables, including the number of columns and their titles, typically remains unchanged. This consistency allows for the formulation of rules to

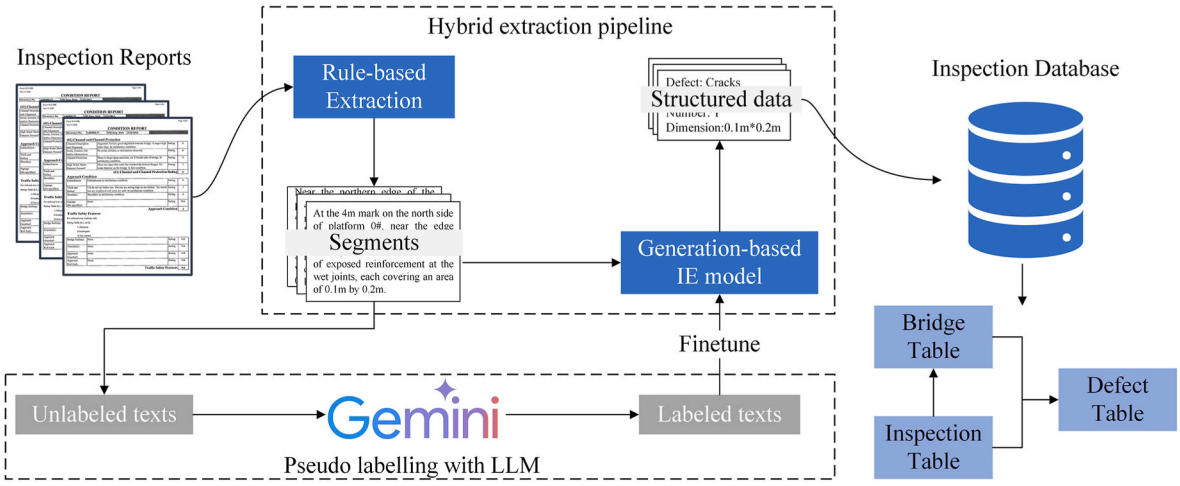

Fig. 1. Overall framework.

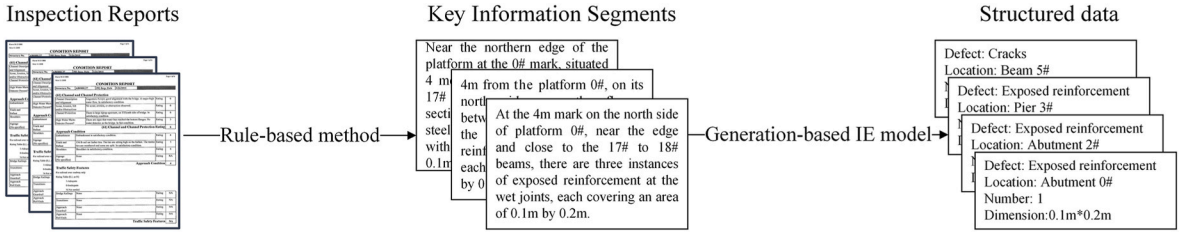

Fig. 2. Hybrid extraction pipeline for bridge inspection reports.

Table 1
Example of defect data table.

Defect description	Component	Bridge	Remark
At the 4m mark on the north side of abutment 0#, near the edge and close to the 17# to 18# beams, there are three instances of exposed reinforcement at the joints, each covering an area of 0.1m by 0.2m.	Load bearing component	Example Bridge	None
…	…	…	…

efficiently identify and extract pertinent tables from the inspection reports.

(1) Browse through the tables contained within the inspection report documents.
(2) Select tables that match in both the number of columns and the titles of those columns.

For instance, the rules designed to extract defect data, as depicted in Table 1, would target tables featuring four columns, with the first column titled "Defect description". The pseudo code outlining the process for extracting defect data is presented in Fig. 3.

The proposed rule-based extraction method can extract the key information segments concerning static data (which includes general

```
Algorithm ExtractDataFromTables
Input: docxDocument
Output: results list

Begin
    results = Initialize an empty list;

    For each table in docxDocument do:
        columnCount = Get the number of columns in the current table;
        firstColumnName = Get the text in the first cell of the first row in the current table;

        If columnCount equals 4 and firstColumnName equals "Defect description" then:
            For each row in the table starting from the second row do:
                rowData = Initialize an empty list;

                For each cell in the row do:
                    Append the text of the cell to rowData;

                Append rowData to results;
            End For
        End If
    End For

    return results;
End
```

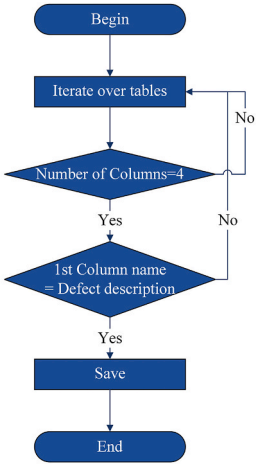

Fig. 3. Pseudo code for extracting defect data tables.

description about the bridge), inspection results and defect data. While static data and inspection results are in a structured format, defect data mainly consists of unstructured textual data, which requires further processing, as shown in Fig. 4.

3.1.2. Generation-based information extraction model

The segments extracted using the proposed rule-based IE model consist of unstructured texts in natural language. To transform these unstructured texts into structured data, this section introduces a generation-based information extraction (IE) model.

As outlined in Section 2.2, IE methods are typically classified as either classification-based or generation-based. In cases of considerable complexity, with overlapping entities and relations, generation-based methods may prove more effective than traditional classification-based approaches. Such methods circumvent the necessity for sophisticated sequence tagging, enabling the direct generation of the desired output with the appropriate prompt. To exemplify, the extraction of structured defect data via traditional IE methods necessitates three stages: named entity recognition (NER), relation extraction (RE), and reconstruction. In contrast, generation-based methods achieve this in a single step, as illustrated in Fig. 5. A further advantage of utilizing a generation-based IE model in our hybrid extraction pipeline is that it can directly employ pseudo labels proposed in this study in an end-to-end manner, obviating the necessity to convert them into complex tagging schemes such as BIO. A further comparison between traditional methods and our approach will be presented in the discussion section. The two advantages render the utilization of generation-based IE models indispensable in this study.

The selection of the generation model has a considerable impact on the performance of our IE pipeline. In an ideal scenario, state-of-the-art LLMs such as ChatGPT, Gemini, or Claude would be the preferred choice. However, in the context of bridge maintenance, financial limitations represent a significant challenge. As illustrated in Table 2, as of August 21, 2024, the prevailing cost for mainstream LLMs is $3 to $5 per million input tokens and $7 to $15 per million output tokens. The exclusive reliance on closed-source LLMs can result in significant financial burden. Furthermore, in certain regions, such as China, access to these closed-source models can be unreliable, with the potential for legal complications. It is therefore beneficial to utilize an open-source generation model.

The recent release of GLM4(GLM Zeng et al., 2024), a powerful open-source language model developed by ZhiPu AI, makes it an ideal candidate for a generation-based IE model, particularly given its native support for both Chinese and English. By leveraging the capabilities of the system, we implemented a one-time prompting strategy to extract all properties in a single round of interaction, as shown in Table 3.

The prompt was structured into three principal sections: a description of the task, a delineation of the requirements for responses, and an illustrative example. The task description provides a succinct and unambiguous account of the problem. It is imperative that the response format requirements and example sections are adhered to in order to guarantee the production of a high-quality model output. This is achieved by providing guidance to the model in terms of its responses, thereby reducing the necessity for post-processing. It is recommended that this prompting structure be employed for all related tasks.

Despite the strong capabilities of GLM4, direct utilization of the model on the task in question does not yield satisfactory results, as demonstrated in Section 4.4. It is therefore evident that further fine-tuning is required to enhance the model's performance in the extraction of defect data. To prepare training data, the text in the labels was incorporated into the prompt and the structured JSON-format defect data was converted into a string, which served as the desired response. This process is illustrated in Fig. 6.

The model was then fine-tuned using the Low-Rank Adaptation (LoRA) technique(Hu Shen et al., 2021). To maximize performance, we conducted a series of experiments to determine the optimal settings. Details on the fine-tuning parameters and experiments are provided in Section 4.4.

Although the labeling scheme for a generation-based model is simpler than that of a classification-based model, the manual labeling process remains labor-intensive. To address this, we introduced a pseudo-labeling method to alleviate the time-consuming burden of manual labeling, as illustrated in Section 3.2.

3.2. Pseudo-labelling with LLMs

Despite the accumulation of extensive text corpora across a range of general domains, the challenge of obtaining high-quality training data for domain-specific tasks persists. In domains such as bridge inspection, it is exceedingly difficult to obtain an existing corpus. To reduce the burden of manual data labelling, we developed a pseudo-labelling method using a LLM to generate high-quality pseudo-labeled. This section outlines the procedures used in this study to generate pseudo.

Using the rule-based extraction method described in Section 3.1.1, we obtained a substantial amount of unlabeled textual data. Then we employed a powerful closed-source LLM, Gemini, developed by Google (Gemini Team Anil et al., 2024) to generate high-quality pseudo-labels for this data. In this process, to further enhance the quality of the pseudo-labeled data, we adopted a few-shot learning prompting strategy (Brown Mann et al., 2020), using a few human-annotated examples to guide the LLM in completing the task. Finally, we assessed the quality of the pseudo labels by comparing them with human-annotated labels to validate the effectiveness of the proposed method. The overall pseudo-labelling process is illustrated in Fig. 7.

The prompt used in the pseudo-labeling process is consistent with that used in the generation-based IE model, as shown in Table 3. We also

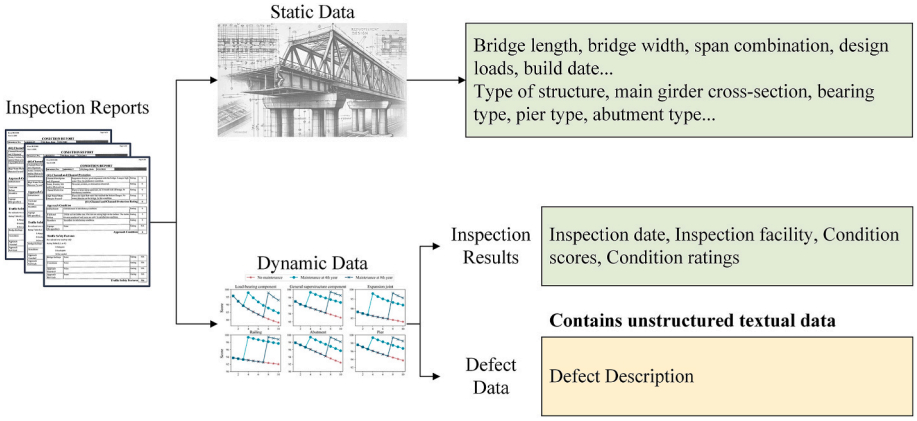

Fig. 4. Content of the inspection reports.

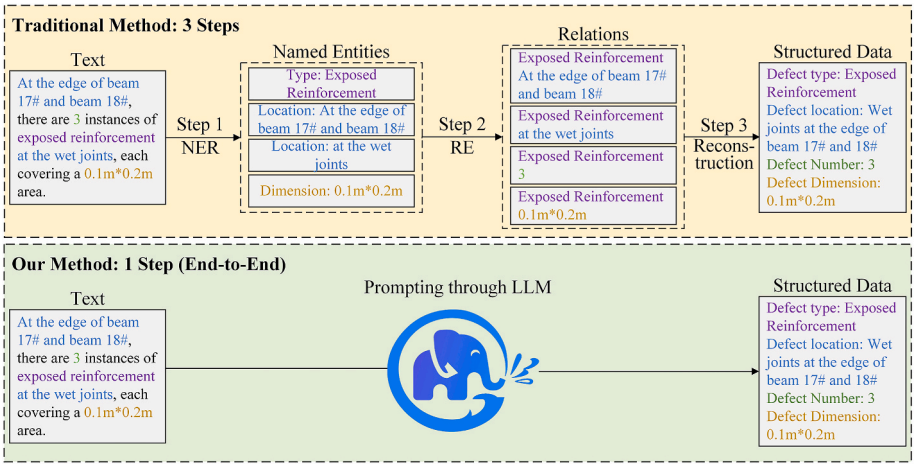

Fig. 5. Comparison between traditional classification-based IE method and generation-based IE method adopted in this study.

Table 2
Price for state-of-the-art LLMs' API.

Model	API cost (per 1 million tokens)	
	Input ($)	Output ($)
Gemini-1.5-Pro	3.500	7.000
gpt-4o	5.000	15.000
Claude 3 Sonnet	3.000	15.000

Table 3
Prompts used for information extraction.

Section	Prompt
Task description	You are a Named Entity Recognition model analyzing Chinese bridge defect descriptions, tasked with extracting key information from each sentence. Complete this task by following the streamlined steps below: 1. Assess Defects: Count the distinct defects described. 2. Extract Information: For each defect, record: - Location: Specific bridge area of the defect. - Type: Kinds of defects. - Dimension: Size, spacing, or extent. - Number: Quantity of defects described.
Response format requirement	For each defect, format your findings as a list of dictionaries, with keys for "defect location", "defect type", "defect dimension", and "defect number", using "Not mentioned" if a detail is not explicitly provided. Ensure your response adheres to this JSON structure: {"defects":[{"key": "value", … }, …]}
Example	Here are some examples: Example X: Text: "{The text describing defects}" Response: {"defects":[{"defect type": "{defect type}", "defect location": "{defect location}", "defect number": "{defect number}", "defect dimension": "{defect dimension}"}]}

designed a series of experiments to explore the selection of examples to include in the prompt, which will be detailed in Section 4.3.

3.3. Bridge inspection database construction

The data extracted using the hybrid extraction pipeline forms the basis for establishing a bridge inspection database. To organize this information effectively, we structured the database into three tables: a bridge table, an inspection table, and a defect table, which corresponds to the static data, inspection results and the defect data illustrated in Fig. 4 respectively. The specific keys used in each of these tables are detailed in Table 4.

The bridge table primarily stores key information about each bridge, including a unique ID, name, location, span length, structural type, and other structural attributes. The inspection table, linked to the bridge table via a foreign key (bridge ID), holds data pertaining to individual inspections. It includes keys such as inspection ID, date, the facility responsible for the inspection, and the results, which encompass condition scores and ratings of the inspected bridges.

The defect table documents detailed information about defects identified on the bridge. It is connected to both the bridge table and the inspection table through foreign keys—bridge ID and inspection ID, respectively—to facilitate ease of querying. Additionally, a 'parent defect ID' is assigned to each defect to monitor its development over time. Attributes in the defect table include defect type, location, number, dimension, and severity, providing a comprehensive overview of each identified defect.

The overall structure of the database is depicted in Fig. 8.

4. Experiments and results

4.1. Data description

We collected real-life inspection reports spanning from 2012 to 2023, which included results from inspections of 99 bridges in China over an 11-year period. Using the hybrid extraction pipeline, we extracted the static data, the inspection results, and the defect data of these bridges.

We then leveraged the extracted defect data, as detailed in Table 1, which comprises over 49,000 data entries, to validate the effectiveness of our proposed generation-based information extraction method. To assess the model's performance, we manually labeled 200 of these entries as ground truth data. The labeled defect data comprises five attributes, as outlined in Table 5. We categorized the labeled data according to two criteria: complexity and completeness. The classification criteria are detailed in Table 11 in the appendix, with the distribution of data across each category shown in Table 6. This subset served as a benchmark to measure the accuracy and reliability of our information extraction process.

As the labeled data are randomly sampled from the original dataset, the distribution of the former closely mirrors that of the latter. As illustrated in Table 6, the majority of defect text descriptions encompass a single defect and frequently lack certain attributes. Nevertheless, the category of multiple and overlapping defects still represents approximately 25% of the total dataset.

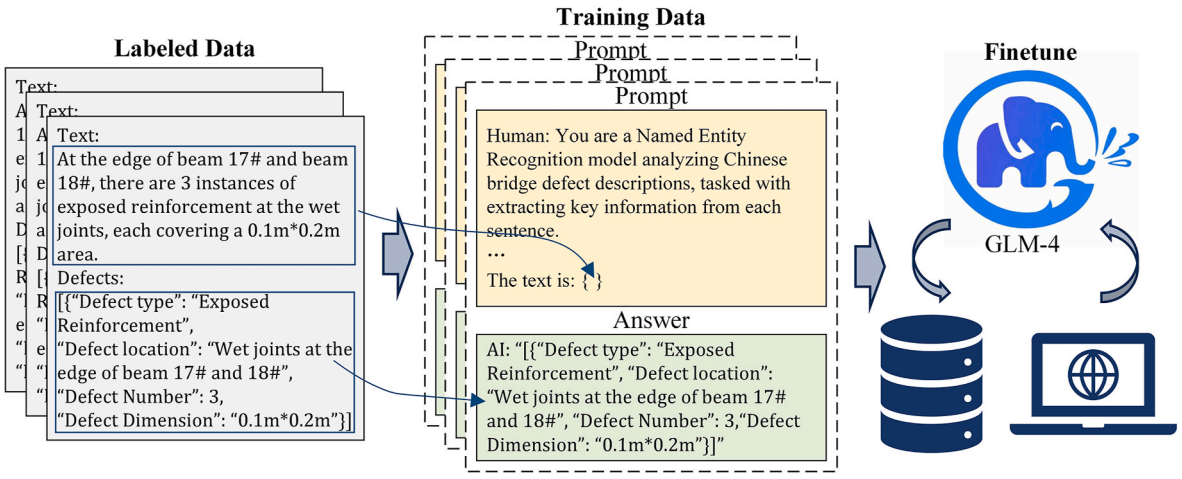

Fig. 6. Preparation of training data.

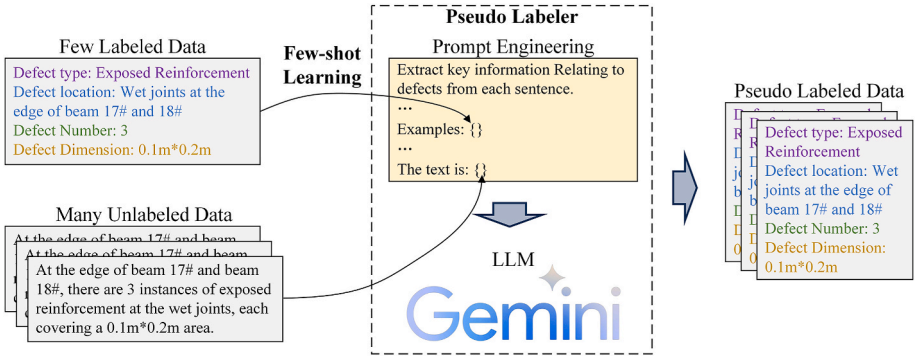

Fig. 7. Pseudo labelling process.

Table 4
Keys of the bridge inspection database.

Table	Primary key	Foreign Key	Attributes
Bridge	Bridge ID	\	Bridge name, Bridge location, Span length, Structural type, Other structural attributes …
Inspection	Inspection ID	Bridge ID	Inspection date, Inspection facility, Condition scores, Condition ratings
Defect	Defect ID	Bridge ID, Inspection ID, Parent Defect ID	Defect type, Defect location, Defect number, Defect dimension, Defect severity

4.2. Metrics in this study

In contrast to conventional classification-based IE techniques, which entail assigning a label to each character in a sentence, the generation-based IE approach employed in this study identifies entities and relations by summarizing the source texts. Accordingly, ROUGE (Recall-Oriented Understudy for Gisting Evaluation) was employed as the principal metric for evaluating the quality of the extracted information. For the reader's convenience, we also calculated traditional character-level precision metrics, which we will refer to here as the character metric.

ROUGE is commonly used in the assessment of text generated by language models, measuring the overlap of various units such as n-grams between the predictions and the ground truth. Given the straightforward nature of the texts under review, we opted for the ROUGE-1 metric, which evaluates the overlap of 1-g between the predictions and the ground-truth labels.

ROUGE provides 3 types of evaluation metrics: P (Precision), R (Recall), and F (F1-score), demonstrated as Eqn. (1) to Eqn. (3):

$$ROUGE-P=\frac{\sum_{s\in\{Ground\ Truth\}}\sum_{gram_1\in s}Count_{match}(gram_1)}{\sum_{s\in\{Prediction\}}\sum_{gram_1\in s}Count(gram_1)} \tag{1}$$

$$ROUGE-R=\frac{\sum_{s\in\{Ground\ Truth\}}\sum_{gram_1\in s}Count_{match}(gram_1)}{\sum_{s\in\{Prediction\}}\sum_{gram_1\in s}Count(gram_1)} \tag{2}$$

$$ROUGE-F=2\times\frac{P\times R}{P+R} \tag{3}$$

In the equations, $Count_{match}(gram_1)$ represents the count of 1-g in the prediction that match the ground truth, while $Count(gram_1)$ is the count of n-grams in the prediction or ground truth. As $ROUGE-F$ provides a more universal view on the results, we choose it as the evaluation metrics. And we use $\overline{ROUGE-F_t}$ to measure the accuracy of generated labels in test t, defined as Eqn. (4).

$$\overline{ROUGE-F_t}=\frac{1}{N_eN_s}\sum_{i_e}^{N_e}\sum_{i_s}^{N_s}ROUGE-F_{i_{s,e}} \tag{4}$$

In Eqn. (4), $ROUGE-F_{i_{s,e}}$ is the ROUGE-F score of the sample s and entity e, while N_e is the number of entities and N_s is the number of samples in the test.

The character metric assesses the accuracy of predictions in relation

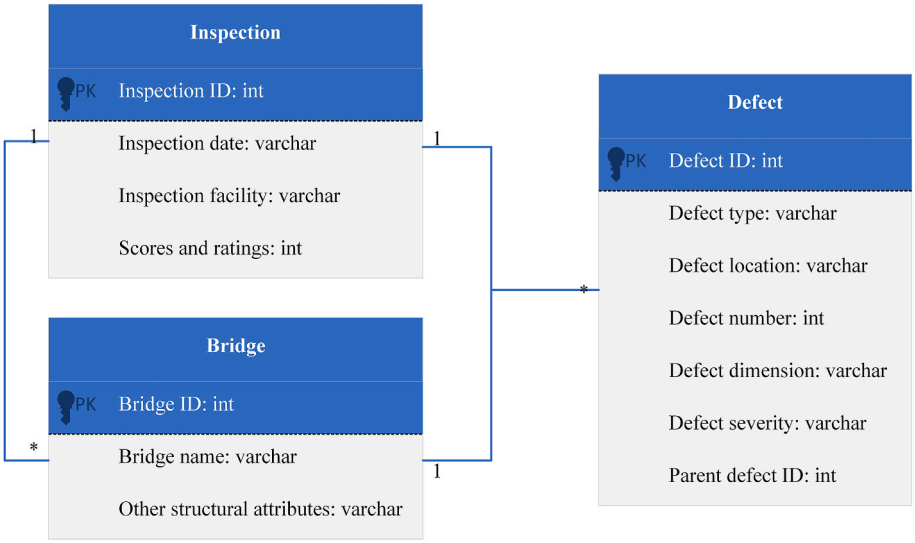

Fig. 8. Structure of the bridge inspection database.

Table 5
Attributes of the defect data.

Attribute	Explanation	Example
Defect Description	The original text description of the defect	At the edge of beam 17# and beam 18#, there are 3 instances of exposed reinforcement at the wet joints, each covering a 0.1m*0.2m area.
Defect Type	Type of the defect described	exposed reinforcement
Defect Location	Where the defect is located in the description	Wet joints at the edge of beam 17# and 18#
Defect Number	How many defects are described	3
Defect Dimension	The dimension of the described defect	0.1m*0.2m

Table 6
Distribution of labeled data across categories.

Number of Data	Single (S)	Multiple, No Overlap (M, N.O.)	Multiple, Overlap (M, O.)	Total
Complete	37	6	5	48
Incomplete	104	3	45	152
Total	141	9	50	200

to the ground truth at the character level. In order to compute this metric, each character in the text is first converted into a unique ID, and a dynamic padding strategy is employed in order to prevent errors that may arise from the use of padding. Subsequently, the F1 score is calculated by comparing the ID lists of the predictions and the ground truth at the character level, as shown in Eqn. (5) to Eqn. (7).

$$\mathrm{F1} = \frac{2 \times \mathrm{precision} \times \mathrm{recall}}{\mathrm{precision} + \mathrm{recall}} \tag{5}$$

$$\mathrm{precision} = \frac{\sum_{i=1}^{k} 1\left(a_{Pi}^{*} = a_{Gi}^{*}\right)}{\sum_{i=1}^{k} 1\left(a_{Pi}^{*} \neq 0\right)} \tag{6}$$

$$\mathrm{recall} = \frac{\sum_{i=1}^{k} 1\left(a_{Pi}^{*} = a_{Gi}^{*}\right)}{\sum_{i=1}^{k} 1\left(a_{Gi}^{*} \neq 0\right)} \tag{7}$$

In Eqn. (6) and Eqn. (7), a_{Pi}^{*} is the ID of character i in the padded prediction ID list, and a_{Gi}^{*} is the ID of character i in the padded ground truth ID list. $\mathbf{1}\left(a_{Pi}^{*} = a_{Ri}^{*}\right)$ is the indicator function, which returns 1 if the condition inside is true and 0 otherwise.

Generally, the ROUGE metric is more robust as it can better cope with little edit difference between 2 sentences with the same semantic meaning (Owczarzak et al., 2012).

4.3. Verification of the pseudo-labelling method

To evaluate the effectiveness of our proposed pseudo-labeling method, we conducted experiments using the manually labeled dataset of 200 entries. We designed six experimental scenarios, varying the number and type of examples included in the prompt, as detailed in Table 7.

The zero-shot prompting strategy serves as a baseline by excluding any examples in the prompt. The 1-shot, 2-shot, 3-shot, and 6-shot prompts each follow a "one-shot" strategy, selecting no more than one example from each data category (which was described in Table 11 in the appendix). The 12-shot prompt, which selects two samples from each data category, represents the few-shot prompting strategy in this experiment.

To reduce the impact of random errors, each experimental scenario

Table 7
Experiment scenario for the verification of the pseudo-labelling method.

Scenario		0-shot	1-shot	2-shot	3-shot	6-shot	12-shot
Number of samples	Complete-S	0	0	0	**1**	**1**	**2**
	Complete-M, N.O.	0	0	0	**1**	**1**	**2**
	Complete-M, O.	0	0	**1**	0	**1**	**2**
	Incomplete-S	0	0	0	0	**1**	**2**
	Incomplete-M, N.O.	0	0	0	0	**1**	**2**
	Incomplete-M, O.	0	**1**	**1**	**1**	**1**	**2**
	Total	**0**	**1**	**2**	**3**	**6**	**12**

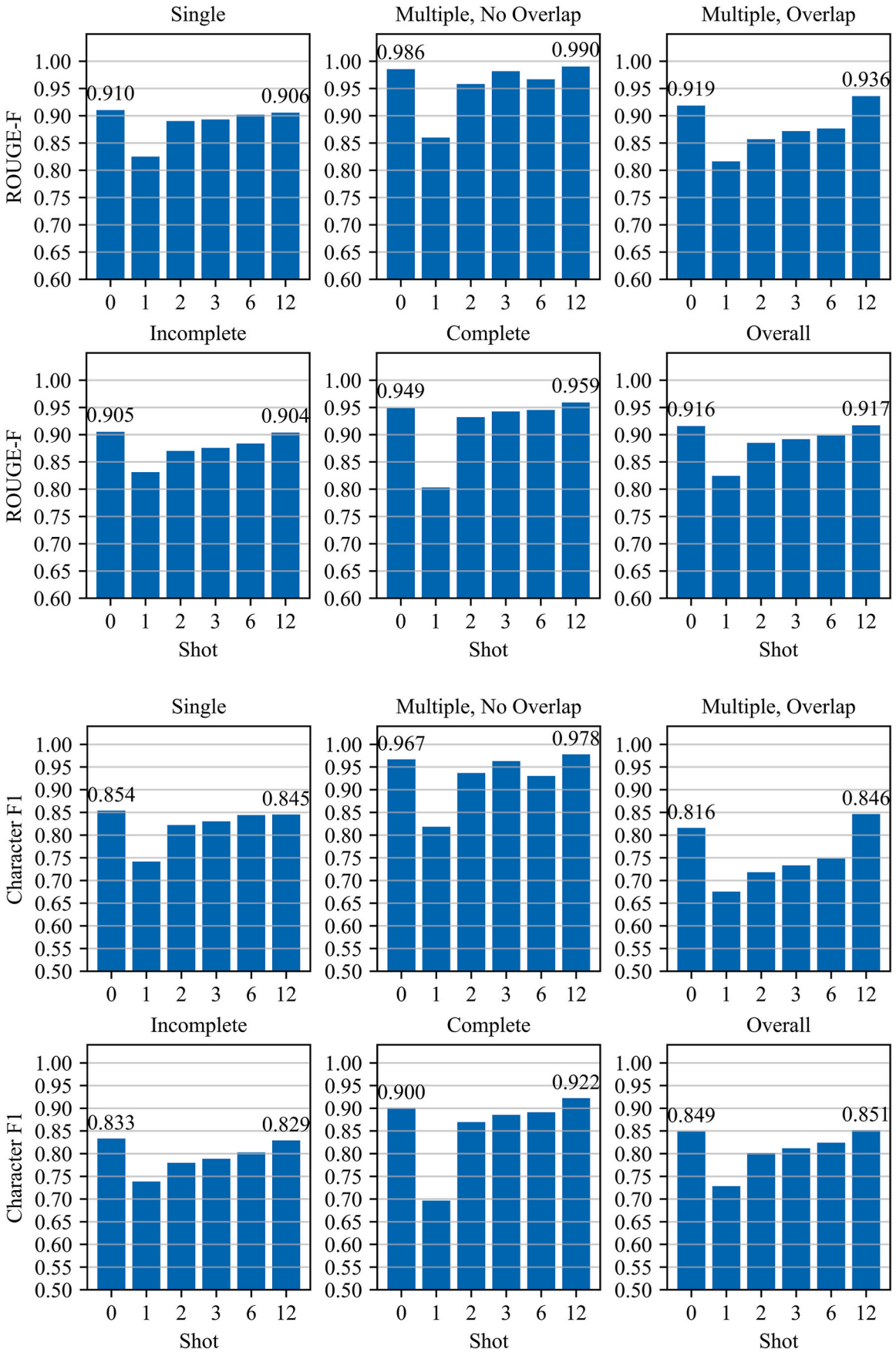

Fig. 9. Evaluation of results for different experiment scenarios.

was tested 10 times. Subsequently, pseudo labels were generated and compared with the ground truth using both the ROUGE and character metric. The results are presented in Fig. 9.

Fig. 9 illustrates that the 12-shot prompting strategy exhibits a slight advantage over the 0-shot approach, while all other prompting strategies demonstrate a notable deficiency. In particular, the 12-shot strategy demonstrates superior performance in the "Multiple, Overlap," "Multiple, No Overlap," and "Complete" data categories, while the zero-shot strategy exhibits a slight advantage in the "Single" and "Incomplete" data categories. This corroborates our hypothesis that "few-shot" prompts tend to outperform "zero-shot" prompts.

A comparison of the results of the 1-shot, 2-shot, 3-shot, and 6-shot strategies, which all belong to “one-shot” prompting, reveals a clear correlation between the number of categories covered by the examples in the prompt and the quality of the outcome. The six-shot strategy, which includes one sample from each data category, outperforms the other three strategies, which lack examples from certain categories. Furthermore, the inferior performance of the one-shot strategies relative to the zero-shot strategy indicates that the exclusion of examples from specific data aspects can have a detrimental impact on the results, potentially introducing bias into the model.

Moreover, Fig. 10 illustrates the assessment of diverse defect entities through the utilization of the 12-shot prompting strategy. In contrast to the uniform conclusions observed in Fig. 9 across both the ROUGE and character metrics, Fig. 10 illustrates some discrepancies. The extraction of the defect location entity achieved the highest score in the ROUGE metric, while the extraction of the defect number entity achieved the highest score in the character metric. Notwithstanding these discrepancies, the "Multiple, No Overlap" category remains the most straightforward for the extraction of all defect entities. Conversely, the extraction of the defect dimension entity was identified as the most challenging task, as reflected in both metrics. But in general, the quality of the pseudo-labels is sufficient for downstream training tasks.

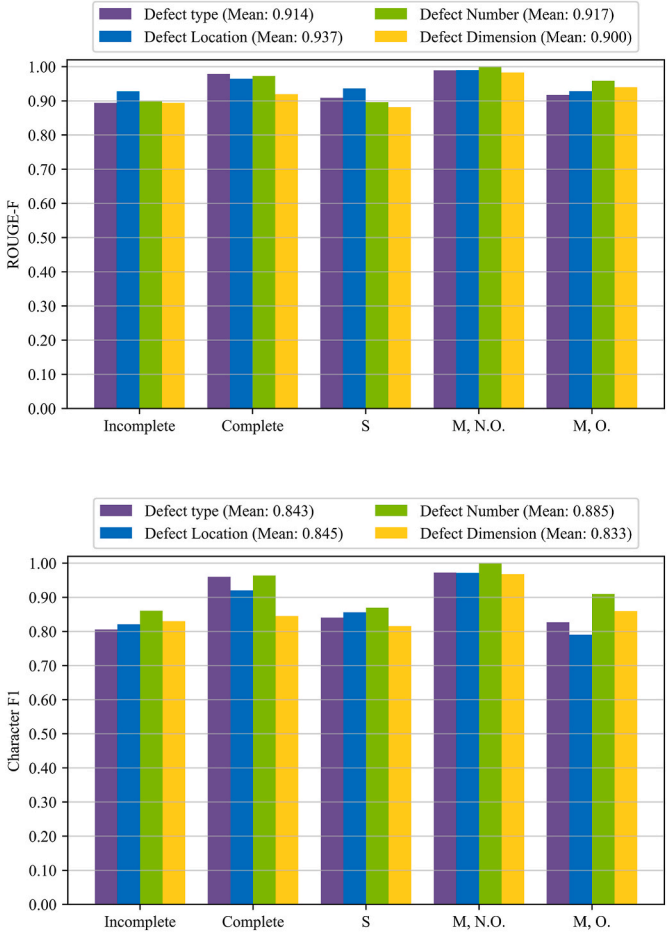

Fig. 10. Evaluation of results across different defect entities for the 12-shot experiment.

4.4. Verification of the generation-based IE model

In this section, we verified the effectiveness of the proposed generation-based IE model on defect data extraction, equipped with the pseudo labels generated by Section 4.3.

Initially, the 12-shot prompting strategy was employed to generate the pseudo-labeled data. Subsequently, the 200 manually labeled entries were divided into training, testing, and validation sets, as detailed in Table 8. Although a larger training dataset is generally associated with superior outcomes, we allocated only 64 entries for training because we intend to incorporate an additional 64 to 6000 pieces of pseudo-labeled data for fine-tuning. To guarantee the most robust possible testing results, a larger portion of the manually labeled data was reserved for testing purposes – specifically, 120 entries, leaving only 64 for fine-tuning.

Subsequently, the ground truth data from the training set was merged with the pseudo-labeled data in order to fine-tune the model, in accordance with the experimental scenarios delineated in Table 9. The ground truth data employed in the training process was maintained as a constant across all scenarios. The objective of this experimental design is to ascertain the optimal number of pseudo-labeled entries for fine-tuning, thereby ensuring that the process is both time- and cost-efficient.

The fine-tuning process was conducted on a server with eight RTX 3090 GPUs, using identical hyper-parameter configurations across all experimental scenarios. The specific hyper-parameters are detailed in Table 12 in the appendix. All models were fine-tuned for 10 epochs, with the overall ROUGE metric used in validation to monitor the training progress. The model from the epoch with the highest validation score was then selected, as an “early stopping” strategy for further testing.

Fig. 11 illustrates the loss and validation scores observed during the fine-tuning process. The figure includes eight sets of lines, each representing a different quantity of training samples used. Within each set, there are two lines: "pure," indicating that only pseudo-labeled data were used for training, and "mixed," indicating that ground-truth data were included in the training mix. From Fig. 11, it is evident that due to the large scale of the model and relatively small number of training data, all models converged within 10 epochs.

The performance of the fine-tuned models across each experimental scenario was tested on the test set, with results shown in Fig. 12. For comparison, the original untuned GLM4 model and Gemini were also tested. As seen in Fig. 12, the model fine-tuned on 2048 mixed training samples achieved the highest precision, surpassing Gemini by 0.017 in the ROUGE metric and 0.029 in the character metric, indicating an optimal training sample size for fine-tuning. Moreover, the fact that the fine-tuned model outperformed the advanced closed-source LLM suggests it effectively captures the underlying patterns in the pseudo-labels, demonstrating its ability to optimally leverage the training data.

Table 8
Division strategy for the labeled data.

Scenario		Train	Validation	Test	Total
Number of samples	Complete-S	16	4	17	37
	Complete-M, N.O.	2	2	2	6
	Complete-M, O.	2	1	2	5
	Incomplete-S	23	4	77	104
	Incomplete-M, N.O.	1	1	1	3
	Incomplete-M, O.	20	4	21	45
	Total	**64**	**16**	**120**	**200**

Table 9
Experiment scenario for the verification of the generation-based IE model.

Scenario	Number of ground truth data	Total number of training data
2.1.1	64	128
2.1.2	0	128
2.2.1	64	256
2.2.2	0	256
2.3.1	64	512
2.3.2	0	512
2.4.1	64	1024
2.4.2	0	1024
2.5.1	64	2048
2.5.2	0	2048
2.6.1	64	4096
2.6.2	0	4096
2.7.1	64	5120
2.7.2	0	5120
2.7.1	64	6144
2.7.2	0	6144

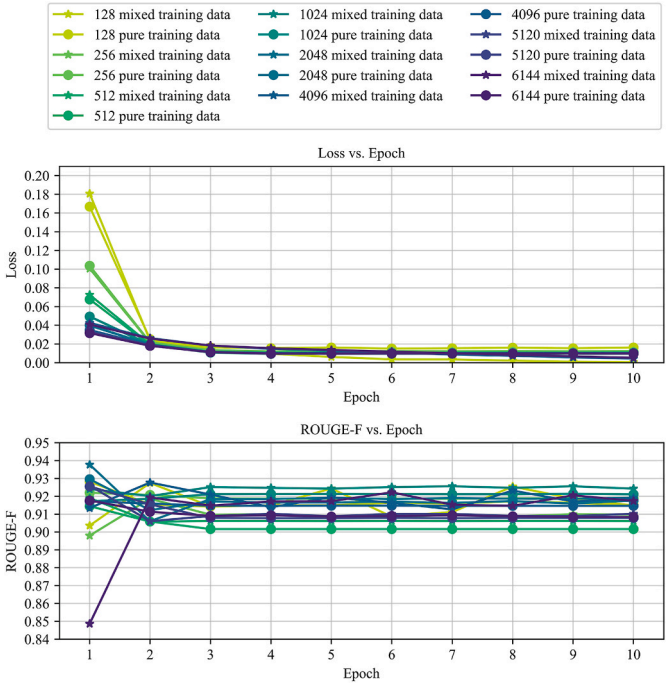

Fig. 11. Training process.

5. Discussions

5.1. Impact of training sample size and ground truth inclusion on IE model precision

The results demonstrate that training the model on 2048 mixed training samples yields the optimal performance. Nevertheless, it is pertinent to question whether this sample size and preparation strategy should be the sole parameters employed.

Fig. 12 demonstrates an interesting phenomenon that the model's precision does not consistently improve as the training sample size increases. Initially, as the sample size grows from 128 to 512, precision decreases. However, beyond 512 samples, precision begins to improve, peaking at 2048 samples before declining again. This pattern is likely due to noise in the training data, particularly in the pseudo-labeled samples. As the sample size increases from 128 to 512, the number of noisy labels also increases, which negatively impacts the model's performance. When the sample size reaches 2,048, the model starts to generalize better, learning patterns that are more resistant to noise,

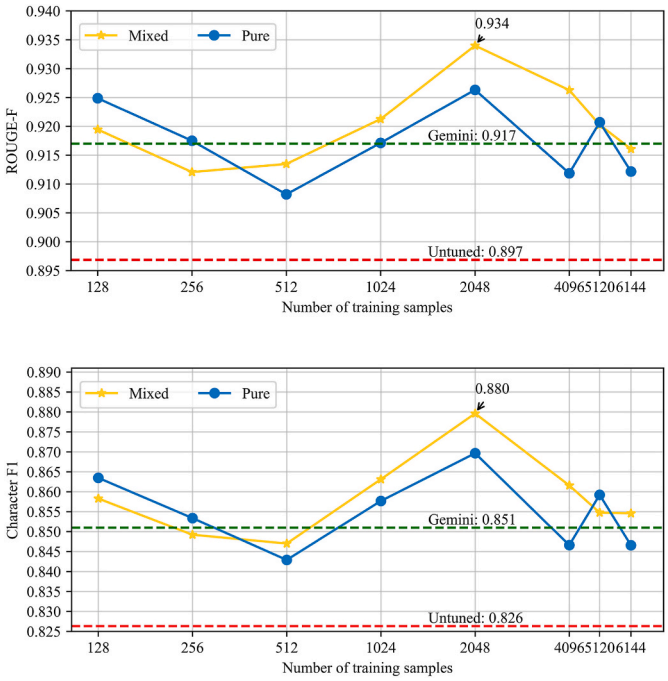

Fig. 12. Evaluation results on the test dataset.

thereby improving precision. However, with further increases in sample size, the noise becomes more dominant, leading to a decline in precision.

This assumption is supported by the performance difference between model precision trained on mixed and pure training samples. In the mixed training samples, 64 ground truth data points are retained, making the model less susceptible to noise, especially when the sample size is under 512. This explains why the precision drop for mixed samples bottoms out at 256 samples, lower than that of pure samples. Additionally, the precision of mixed samples consistently outperforms that of pure samples, as the inclusion of ground truth data enhances the model's robustness to noise.

This suggests that the optimal number of training samples is closely tied to the inclusion of ground truth data. If your budget is limited, preventing the generation of large numbers of pseudo-labels and you lack the resources to provide manually-labeled ground truth data, it is advisable to opt for a smaller training sample size (e.g., 128) using only pseudo labels. This approach can still outperform the closed-source LLM used to generate the pseudo labels. However, if conditions allow, combining a small amount of manually labeled data with approximately 2000 pseudo-labeled samples will yield the best performance.

5.2. Comparison between the proposed IE model and a classification-based IE model

To further demonstrate the superiority of the proposed method, we also trained a classical classification-based IE model, the BERT-BILSTM-CRF model (Dai Wang et al., 2019), for comparison. The defect data extraction process using this model is consistent with the traditional method shown in Fig. 5. First, we re-labeled the 200 data entries using the BIO tagging scheme to train an NER model for entity extraction from the defect descriptions. Next, we labeled the entities in BIO format to create entity-relation triplets, which were used to train a RE model. Additionally, we developed automatic conversion functions to transform the pseudo labels into BIO labels and entity-relation triplets. Finally, the RE model's predictions were reconstructed to produce the JSON-format extraction results, as shown in Table 3. For the ease of comparison, we trained the BERT-BILSTM-CRF model on the 64 manually labeled dataset and the 2048 mixed dataset (which is also the best performing

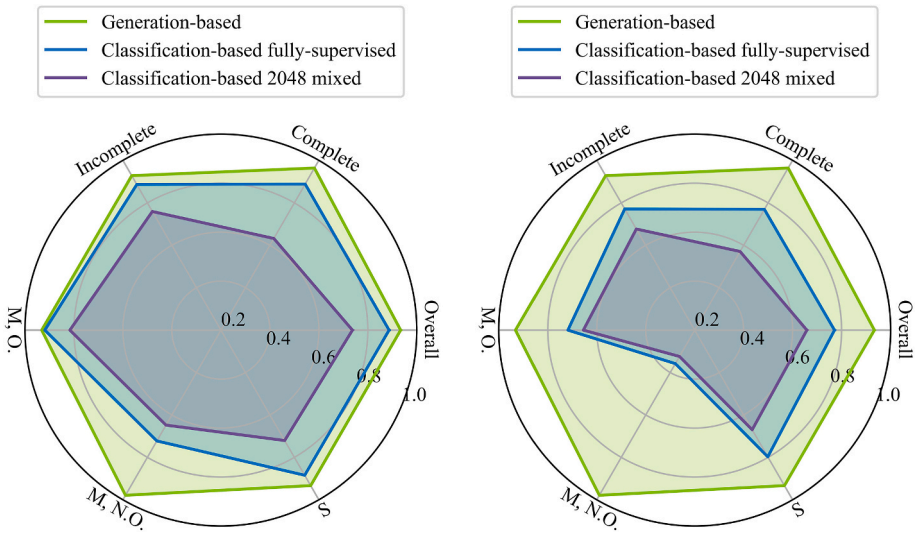

Fig. 13. Comparison of our generation-based IE model and the classification-based model on the overall extraction.

dataset for the generation-based IE model). The methods and metrics used to evaluate the BERT-BILSTM-CRF model's results are consistent with those used for the proposed model.

Fig. 13 compares the performance of our proposed method (the generation-based IE model fine-tuned on 2048 mixed training samples) with two classification-based IE methods: one trained on 64 manually labeled data (referred to as "Classification-based fully-supervised" in the figure) and the other trained on the same 2048 mixed training samples (referred to as "Classification-based 2048 mixed" in the figure). The comparison is made across different categories of test data as well as the overall test dataset.

First, it is evident that the proposed method outperforms both classification-based models across all data categories. Notably, when the text involves multiple defects, the generation-based model achieves nearly perfect accuracy in both metrics, whereas the classification-based models perform poorly. This discrepancy arises because the traditional BIO and entity-relation triplet tagging scheme used in classification-based models struggles with sentences where there are multiple defects, and the model cannot distinguish which entity belongs to which defect.

Interestingly, the model trained on 2048 mixed training samples performed worse than the model trained on just 64 manually labeled data. We believe this is because classification-based models are more sensitive to noise than generation-based models. First, there is inherent noise in the pseudo-labeled data. Then, during the conversion of pseudo-labeled data to BIO and entity-relation triplet formats, additional noise was introduced. This accumulated noise likely hindered the classification-based model's ability to learn proper patterns during training. This finding further supports the use of a generation-based model in our hybrid IE pipeline, as these models are more robust to noise and eliminate the need for converting pseudo labels, thereby avoiding the introduction of additional noise into the training data.

Figs. 14–17 compare the extraction performance for the four defect entities: defect type, defect location, defect number, and defect dimension. The conclusions from these figures align with those from Fig. 13, further confirming the effectiveness of our proposed method.

Overall, the comparison between our proposed method and traditional approaches highlights the advantages and convenience of our hybrid IE pipeline, enhanced with LLM-based pseudo-labeling. Table 10 summarizes these benefits, showing that despite higher computational costs, the proposed method surpasses traditional classification methods in terms of ease of data preparation, effective utilization of pseudo-labeled data, and extraction precision.

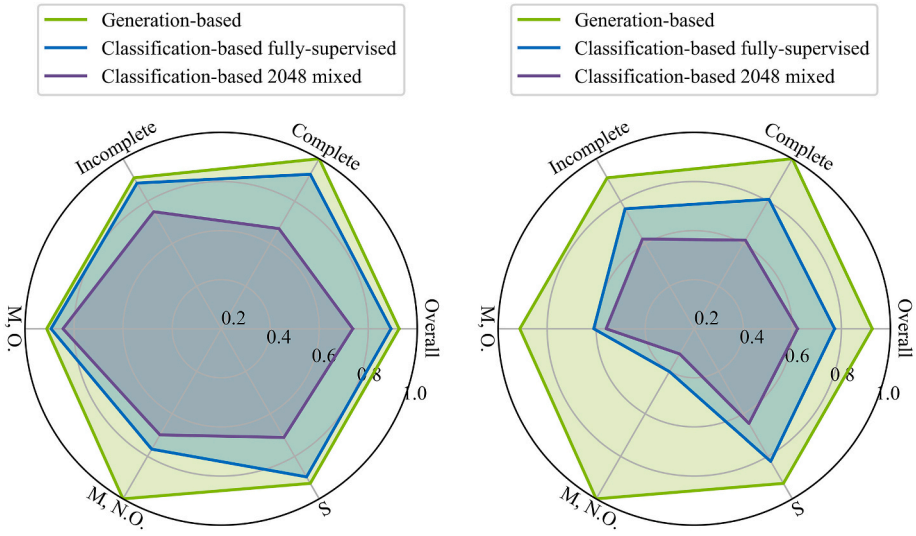

Fig. 14. Comparison of our generation-based IE model and the classification-based model on extraction of defect type.

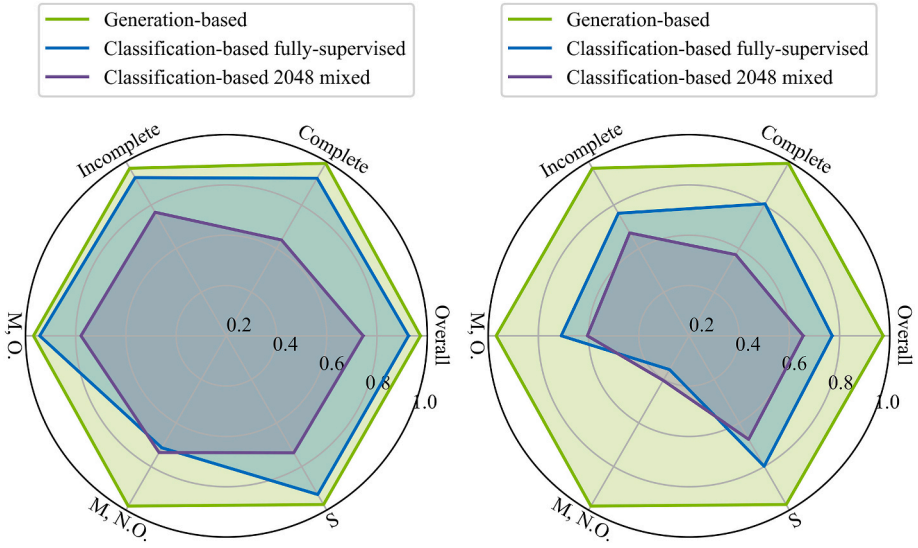

Fig. 15. Comparison of our generation-based IE model and the classification-based model on extraction of defect location.

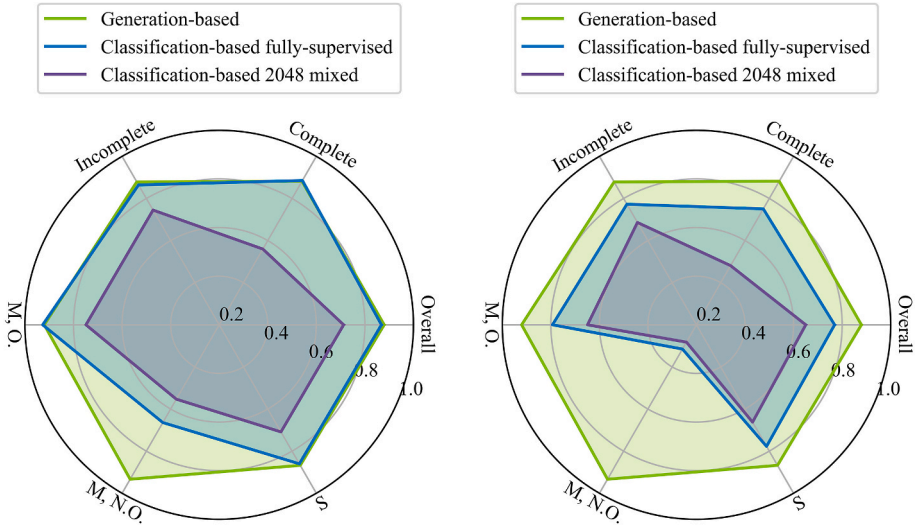

Fig. 16. Comparison of our generation-based IE model and the classification-based model on extraction of defect number.

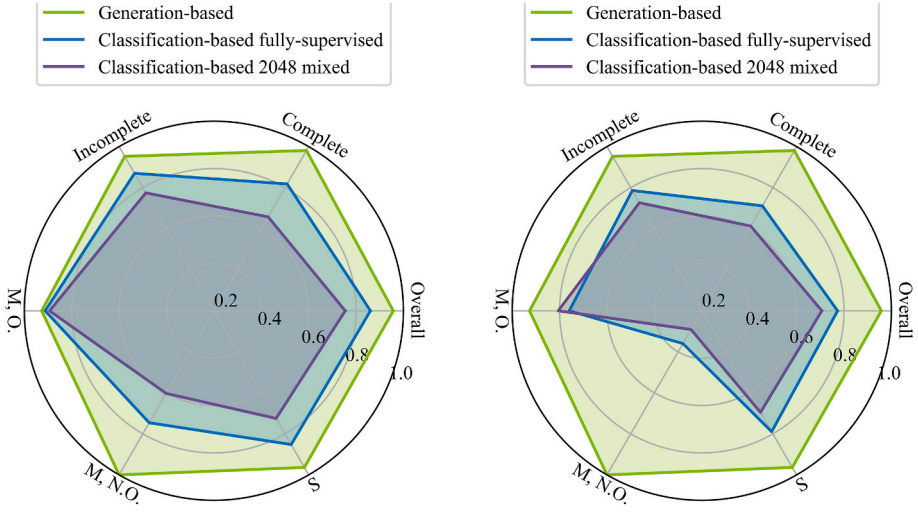

Fig. 17. Comparison of our generation-based IE model and the classification-based model on extraction of defect dimension.

Table 10
Summary Comparison of Proposed vs. Traditional Methods.

Aspect	The Proposed Method	The Traditional Method
Time consumed for preparing the 200 manually labeled data	**1 h**	4 h
End to end utilization of the pseudo-labeled data	**Yes**	No
Computational cost for training	High	**Low**
Precision of extraction	**High**	Low

6. Conclusions

This study introduces a novel bridge inspection database construction method based on LLM-assisted information extraction. Initially, we employed a pseudo-labeling technique with a closed-source LLM to generate high-quality training data for downstream fine-tuning. We then introduced a hybrid extraction method that combines rule-based and intelligent IE techniques to convert unstructured bridge inspection reports into structured data. The intelligent IE model's generation component was fine-tuned using pseudo-labeled data to enhance its performance. Finally, we established a structured bridge inspection database to systematically store the extracted data. Validated using real-life inspection reports from 99 bridges over a decade, the study yields several key conclusions.

1. By leveraging few-shot prompting strategies with examples covering all data categories, the proposed pseudo-labeling method can generate high-quality data, achieving extraction precision above 0.9 across all defect entities as evaluated by the ROUGE metric.
2. The generation-based IE model effectively leverages the pseudo-labels provided by the sophisticated closed-source LLM, accurately capturing the underlying patterns. As a result, the model fine-tuned on approximately 2000 mixed training samples outperforms the closed-source LLM by 3%, demonstrating the robustness and effectiveness of our approach.
3. Thanks to its robustness to noise, the proposed method can achieve better results than the sophisticated closed-source LLM even when fine-tuned on a small number of pseudo-label-only samples, demonstrating its accessibility and effectiveness with a low threshold for utilization.
4. By utilizing a simpler labelling scheme, the proposed method demonstrates superior performance over traditional approaches in defect data extraction, offering significant advantages in ease of data preparation (with a 75% reduction in time), effective utilization of pseudo-labeled data, and extraction precision (with a 5% increase in precision according to the ROUGE metric and a 14% increase according to the character metric).

Nevertheless, it is essential to acknowledge the limitations of this study. The extracted entities require additional processing to ensure uniformity in naming conventions. Future work could focus on improving the post-processing of extracted defect data, including named entity alignment and other techniques, to enhance the quality of the extracted information.

Funding statement

This paper is supported by the National Natural Science Foundation of China [52278313, 52411540031], the Project to Attract Foreign Experts [G2023133018L], the Technology Cooperation Project of Shanghai Qi Zhi Institute Cooperation [SQZ202310].

CRediT authorship contribution statement

Chenhong Zhang: Writing – original draft, Visualization, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Xiaoming Lei:** Writing – review & editing. **Ye Xia:** Writing – review & editing, Supervision, Project administration, Funding acquisition. **Limin Sun:** Project administration, Funding acquisition.

Declaration of generative AI and AI-assisted technologies in the writing process

During the preparation of this work the authors used ChatGPT in order to improve the language. After using this tool/service, the authors reviewed and edited the content as needed and take full responsibility for the content of the publication.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The authors do not have permission to share data.

Appendix

Table 11
Classification criteria for the labeled dataset

Category		Explanation	Example
Completeness	Complete	The defect description includes all the attributes.	Text: *The north cantilever on the east side of the cap beam of pier No. 10 has exposed reinforcement at the location corresponding to beam No. 1, with an area of 0.2m × 0.1m.* Defects: [{defect type: *exposed reinforcement*, defect location: *The north cantilever on the east side of the cap beam of pier No. 10, at the location corresponding to beam No. 1*, defect number: *1*, defect dimension: *0.2m × 0.1m*}]
	Incomplete	The defect description does not include all the attributes	Text: *There are 3 transverse cracks near the location of pier No. 1 on the bottom of beam No. 3–2.* Defects: [{defect type: *transverse crack*, defect location: *near the location of pier No. 1 on the bottom of beam No. 3–2*, defect number: *3*, defect dimension: *Not Mentioned*}]

(*continued on next page*)

Table 11 (*continued*)

Category		Explanation	Example
Complexity	Single (S)	There is only 1 defect described in the text.	Text: *There are 3 transverse cracks near the location of pier No. 1 on the bottom of beam No. 3–2.* Defects: [{defect type: *transverse crack*, defect location: *near the location of pier No. 1 on the bottom of beam No. 3–2*, defect number: *3*, defect dimension: *Not Mentioned*}]
	Multiple No Overlap (M, N.O.)	There are multiple defects in the text. The attributes do not overlap with each other.	Text: *There are 4 cracks on the north web of beam No. 2–5 within a 2.5-m range from pier No. 2,* $\delta \leq 0.1$ mm. *On the south web of beam No. 2–5, there are 9 cracks within a 3-m range from pier No. 2,* $\delta \leq 0.2$ mm. Defects: [{defect type: *crack*, defect location: *on the north web of beam No. 2–5 within a 2.5-m range from pier No. 2*, defect number: *4*, defect dimension: $\delta \leq 0.1$ mm }, [{defect type: *crack*, defect location: *On the south web of beam No. 2–5 within a 3-m range from pier No. 2*, defect number: *9*, defect dimension: $\delta \leq 0.2$ mm }]
	Multiple Overlap (M, O.)	There are multiple defects in the text. The attributes overlap with each other.	Text: *The bottom of slab beam No. 3–26 has 1 1.5m long longitudinal crack with efflorescence at the end near pier No. 2, and 2 1.5m long transverse cracks at the end near abutment No. 3.* Defects: [{defect type: *longitudinal crack with efflorescence*, defect location: ***The bottom of slab beam No. 3–26***, at the end near pier No. 2, defect number: *1*, defect dimension: *1.5m long* }, [{defect type: *transverse cracks*, defect location: ***The bottom of slab beam No. 3–26***, at the end near abutment No. 3, defect number: *2*, defect dimension: *1.5m long* }]

Table 12
Finetune Settings

Parameter Name	Value
Epochs	10
Early Stopping Patience	5
Train Batch Size	2
Gradient Accumulation Steps	4
Fp16 Training	Yes
LoRA-r	8
LoRA-alpha	32
LoRA-dropout	0.1
Deepspeed Optimization Stage	3

References

Brown, T., Mann, B., et al., 2020. Language models are few-shot learners. In: Advances in Neural Information Processing Systems. Curran Associates, Inc., pp. 1877–1901. In: https://proceedings.neurips.cc/paper/2020/hash/1457c0d6bfcb4967418bfb8ac142f64a-Abstract.html. (Accessed 5 April 2024)

Chien, J.-T., Chen, C.-C., 2023. Collaborative pseudo labeling for prompt-based learning. In: 2023 ASIA PACIFIC SIGNAL AND INFORMATION PROCESSING ASSOCIATION ANNUAL SUMMIT AND CONFERENCE, APSIPA ASC. IEEE, New York, pp. 51–56. https://doi.org/10.1109/APSIPAASC58517.2023.10317441.

Dai, Z., Wang, X., et al., 2019. Named entity recognition using BERT BiLSTM CRF for Chinese electronic health records. In: 2019 12th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics. CISP-BMEI), pp. 1–5. https://doi.org/10.1109/CISP-BMEI48845.2019.8965823.

Feng, D.-C., Wang, W.-J., et al., 2023. Condition assessment of highway bridges using textual data and natural language processing- (NLP-) based machine learning models. Struct. Control Health Monit. 2023, e9761154. https://doi.org/10.1155/2023/9761154.

Gemini Team, Anil, R., et al., 2024. Gemini: a family of highly capable multimodal models. https://doi.org/10.48550/arXiv.2312.11805.

Gilardi, F., Alizadeh, M., et al., 2023. ChatGPT outperforms crowd workers for text-annotation tasks. Proc. Natl. Acad. Sci. USA 120, e2305016120. https://doi.org/10.1073/pnas.2305016120.

Glm, T., Zeng, A., et al., 2024. ChatGLM: a family of large language models from GLM-130B to GLM-4 all tools. https://doi.org/10.48550/arXiv.2406.12793.

Grishman, R., 2015. Information extraction. IEEE Intell. Syst. 30, 8–15. https://doi.org/10.1109/MIS.2015.68.

Hsu, I.-H., Huang, K.-H., et al., 2022. DEGREE: a data-efficient generation-based event extraction model. In: Carpuat, M., de Marneffe, M.-C., Meza Ruiz, I.V. (Eds.), Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Association for Computational Linguistics, Seattle, United States, pp. 1890–1908. https://doi.org/10.18653/v1/2022.naacl-main.138.

Hu, E.J., Shen, Y., et al., 2021. LoRA: low-rank adaptation of large language models. https://doi.org/10.48550/arXiv.2106.09685.

Huang, Z., Xu, W., et al., 2015. Bidirectional LSTM-CRF models for sequence tagging. https://doi.org/10.48550/arXiv.1508.01991.

Huizinga, W., Kruithof, M., et al., 2023. Efficient transfer by robust label selection and learning with pseudo-labels. In: 2023 IEEE INTERNATIONAL CONFERENCE ON IMAGE PROCESSING. ICIP, IEEE, New York, pp. 2660–2664. https://doi.org/10.1109/ICIP49359.2023.10222699.

Jiang, Z., Xu, W., et al., 2020. Generalizing natural language analysis through span-relation representations. In: Jurafsky, D., Chai, J., Schluter, N., Tetreault, J. (Eds.), Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics, pp. 2120–2133. https://doi.org/10.18653/v1/2020.acl-main.192. Online.

Kim, M., Kang, P., 2022. Text embedding augmentation based on retraining with pseudo-labeled adversarial embedding. IEEE Access 10, 8363–8376. https://doi.org/10.1109/ACCESS.2022.3142843.

Lai, L., Dong, Y., et al., 2024. Synergetic-informed deep reinforcement learning for sustainable management of transportation networks with large action spaces. Autom. ConStruct. 160, 105302. https://doi.org/10.1016/j.autcon.2024.105302.

Lei, X., Dong, Y., et al., 2023. Sustainable life-cycle maintenance policymaking for network-level deteriorating bridges with a convolutional autoencoder–structured reinforcement learning agent. J. Bridge Eng. 28, 04023063. https://doi.org/10.1061/JBENF2.BEENG-6159.

Lei, X., Sun, M., et al., 2024. Unsupervised vision-based structural anomaly detection and localization with reverse knowledge distillation. Struct. Control Health Monit. 2024, 8933148. https://doi.org/10.1155/2024/8933148.

Levy, O., Seo, M., et al., 2017. Zero-shot relation extraction via reading comprehension. In: Levy, R., Specia, L. (Eds.), Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017). Association for Computational Linguistics, Vancouver, Canada, pp. 333–342. https://doi.org/10.18653/v1/K17-1034.

Li, T., Alipour, M., et al., 2021. Mapping textual descriptions to condition ratings to assist bridge inspection and condition assessment using hierarchical attention. Autom. ConStruct. 129, 103801. https://doi.org/10.1016/j.autcon.2021.103801.

Li, T., Harris, D., 2019. Automated construction of bridge condition inventory using natural language processing and historical inspection reports, in: nondestructive Characterization and Monitoring of Advanced Materials. Aerospace, Civil Infrastructure, and Transportation XIII, SPIE 206–213.

Li, X., Feng, J., et al., 2020. A unified MRC framework for named entity recognition. In: Jurafsky, D., Chai, J., Schluter, N., Tetreault, J. (Eds.), Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics, pp. 5849–5859. https://doi.org/10.18653/v1/2020.acl-main.519. Online.

Lin, J., Hu, Z., et al., 2016. A natural-language-based approach to intelligent data retrieval and representation for cloud BIM. Computer Aided Civil Eng 31, 18–33. https://doi.org/10.1111/mice.12151.

Liu, K., El-Gohary, N., 2020. Fusing data extracted from bridge inspection reports for enhanced data-driven bridge deterioration prediction: a hybrid data fusion method. J. Comput. Civ. Eng. 34, 04020047. https://doi.org/10.1061/(ASCE)CP.1943-5487.0000921.

Malik, U., Bernard, S., et al., 2024. Pseudo-labeling with large language models for multi-label emotion classification of French tweets. IEEE Access 12, 15902–15916. https://doi.org/10.1109/ACCESS.2024.3354705.

Owczarzak, K., Conroy, J.M., et al., 2012. An assessment of the accuracy of automatic evaluation in summarization. In: Conroy, J.M., Dang, H.T., Nenkova, A., Owczarzak, K. (Eds.), Proceedings of Workshop on Evaluation Metrics and System Comparison for Automatic Summarization. Association for Computational Linguistics, Montréal, Canada, pp. 1–9. https://aclanthology.org/W12-2601. (Accessed 22 August 2024).

Sazzed, S., 2021. Improving sentiment classification in low-resource Bengali language utilizing cross-lingual self-supervised learning. In: Metais, E., Meziane, F., Horacek, H., Kapetanios, E. (Eds.), NATURAL LANGUAGE PROCESSING AND INFORMATION SYSTEMS (NLDB 2021). Springer International Publishing Ag, Cham, pp. 218–230. https://doi.org/10.1007/978-3-030-80599-9_20.

Thompson, P.D., Small, E.P., et al., 1998. The pontis bridge management system. Struct. Eng. Int. 8, 303–308. https://doi.org/10.2749/101686698780488758.

Wang, N., Issa, R.R.A., et al., 2022. NLP-based query-answering system for information extraction from building information models. J. Comput. Civ. Eng. 36, 04022004. https://doi.org/10.1061/(ASCE)CP.1943-5487.0001019.

Wen, C., Chen, T., et al., 2021. Medical named entity recognition from un-labelled medical records based on pre-trained language models and domain dictionary. Data Intell 3, 402–417. https://doi.org/10.1162/dint_a_00105.

Wu, L.-T., Lin, J.-R., et al., 2022. Rule-based information extraction for mechanical-electrical-plumbing-specific semantic web. Autom. ConStruct. 135, 104108. https://doi.org/10.1016/j.autcon.2021.104108.

Xia, Y., Lei, X., et al., 2022. A data-driven approach for regional bridge condition assessment using inspection reports. Struct. Control Health Monit. 29, e2915.

Yan, H., Gui, T., et al., 2021. A unified generative framework for various NER subtasks. In: Zong, C., Xia, F., Li, W., Navigli, R. (Eds.), 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). Association for Computational Linguistics, pp. 5808–5822. https://doi.org/10.18653/v1/2021.acl-long.451. Online.

Yan, H., Sun, Y., et al., 2023. UTC-IE: a unified token-pair classification architecture for information extraction. In: Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Association for Computational Linguistics, Toronto, Canada, pp. 4096–4122. https://doi.org/10.18653/v1/2023.acl-long.226.

Yin, M., Tang, L., et al., 2024. A deep natural language processing-based method for ontology learning of project-specific properties from building information models. Comput. Aided Civ. Infrastruct. Eng. 39, 20–45. https://doi.org/10.1111/mice.13013.

Yu, J., Bohnet, B., et al., 2020. Named entity recognition as dependency parsing. In: Jurafsky, D., Chai, J., Schluter, N., Tetreault, J. (Eds.), Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics, pp. 6470–6476. https://doi.org/10.18653/v1/2020.acl-main.577. Online.

Zhang, J., El-Gohary, N.M., 2016. Semantic NLP-based information extraction from construction regulatory documents for automated compliance checking. J. Comput. Civ. Eng. 30, 04015014. https://doi.org/10.1061/(ASCE)CP.1943-5487.0000346.

Zhang, F., Chan, A.P.C., et al., 2022. Integrated applications of building information modeling and artificial intelligence techniques in the AEC/FM industry. Autom. ConStruct. 139, 104289. https://doi.org/10.1016/j.autcon.2022.104289.

Zhang, H., Li, L., et al., 2023a. PGLR: pseudo graph and label reuse for entity relation extraction. In: 2023 INTERNATIONAL JOINT CONFERENCE ON NEURAL NETWORKS, IJCNN. IEEE, New York. https://doi.org/10.1109/IJCNN54540.2023.10191693.

Zhang, D., Li, H., et al., 2023b. Task relation distillation and prototypical pseudo label for incremental named entity recognition. In: PROCEEDINGS OF THE 32ND ACM INTERNATIONAL CONFERENCE ON INFORMATION AND KNOWLEDGE MANAGEMENT, CIKM. Assoc Computing Machinery, New York, pp. 3319–3329. https://doi.org/10.1145/3583780.3615075, 2023.

Zheng, S., Wang, F., et al., 2017. Joint extraction of entities and relations based on a novel tagging scheme. In: Barzilay, R., Kan, M.-Y. (Eds.), Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Association for Computational Linguistics, Vancouver, Canada, pp. 1227–1236. https://doi.org/10.18653/v1/P17-1113.

Zheng, Z., Lu, X.-Z., et al., 2022. Pretrained domain-specific language model for natural language processing tasks in the AEC domain. Comput. Ind. 142, 103733. https://doi.org/10.1016/j.compind.2022.103733.

Zheng, Z., Zhou, Y.-C., et al., 2022. Knowledge-informed semantic alignment and rule interpretation for automated compliance checking. Autom. ConStruct. 142, 104524. https://doi.org/10.1016/j.autcon.2022.104524.

Zheng, Z., Zhou, Y.-C., et al., 2024. A text classification-based approach for evaluating and enhancing the machine interpretability of building codes. Eng. Appl. Artif. Intell. 127, 107207. https://doi.org/10.1016/j.engappai.2023.107207.