Front. Eng. Manag. 2023, 10(2): 237–249
https://doi.org/10.1007/s42524-021-0179-8

RESEARCH ARTICLE

Qiqi ZHANG, Cong XUE, Xing SU, Peng ZHOU, Xiangyu WANG, Jiansong ZHANG

Named entity recognition for Chinese construction documents based on conditional random field

Abstract Named entity recognition (NER) is essential in many natural language processing (NLP) tasks such as information extraction and document classification. A construction document usually contains critical named entities, and an effective NER method can provide a solid foundation for downstream applications to improve construction management efficiency. This study presents a NER method for Chinese construction documents based on conditional random field (CRF), including a corpus design pipeline and a CRF model. The corpus design pipeline identifies typical NER tasks in construction management, enables word-based tokenization, and controls the annotation consistency with a newly designed annotating specification. The CRF model engineers nine transformation features and seven classes of state features, covering the impacts of word position, part-of-speech (POS), and word/character states within the context. The F1-measure on a labeled construction data set is 87.9%. Furthermore, as more domain knowledge features are infused, the marginal performance improvement of including POS information will decrease, leading to a promising research direction of POS customization to improve NLP performance with limited data.

Keywords NER, NLP, Chinese language, construction document

Received February 18, 2021; accepted July 26, 2021

Qiqi ZHANG, Cong XUE, Xing SU (✉)
College of Civil Engineering and Architecture, Zhejiang University, Hangzhou 310058, China
E-mail: xsu@zju.edu.cn

Peng ZHOU
School of Management Science and Engineering, Central University of Finance and Economics, Beijing 100081, China

Xiangyu WANG
School of Design and the Built Environment, Curtin University, Perth, Western Australia 6845, Australia

Jiansong ZHANG
School of Construction Management Technology, Purdue University, West Lafayette, IN 47907, USA

This work is supported by the National Natural Science Foundation of China (Grant No. 71971196).

1 Introduction

Named entity recognition (NER) is critical in many natural language processing (NLP) tasks such as automatic text summarization, machine translation, information retrieval, and question answering (Manning and Schutze, 1999). According to Goyal et al. (2018), "a named entity (NE) is a word form that recognizes the elements with similar properties from a collection of different elements". NER tasks in general domains aim to recognize common NEs such as persons, locations, and organizations. By contrast, NER tasks in a specific area (e.g., health care (Jauregi Unanue et al., 2017)) aim to recognize domain-specific NEs (e.g., drug names, test names, and treatments). A significant number of NER efforts have been made to support various applications, such as tweet analysis (Liu and Zhou, 2013), electronic health record management (Quimbaya et al., 2016), and agricultural document analysis (Gangadharan and Gupta, 2020).

In the construction domain, a large amount of information is recorded in an unstructured textual form in many different construction documents, such as contracts, building codes, correspondences, daily logs, and supervisory reports. A construction document usually contains domain-specific NEs (e.g., building component, material, and equipment). Recognizing NEs is essential to enable efficient analysis on documents. Examples include construction site accident analysis (Tixier et al., 2016), information retrieval (Le et al., 2018), automated content analysis (Zhang and El-Gohary, 2016; Zhou and El-Gohary, 2017), and compliance checking (Zhou and El-Gohary, 2016; Xu and Cai, 2020). Most of the existing studies tried to identify NEs by making rules or building gazetteers, which are tedious and time-consuming.

Currently, the subject is still an open research area, and several factors contribute to the challenges of building a NER method for Chinese construction documents. They can be mainly categorized into domain-specific factors, grammar factors, and entity type factors.

Domain-specific factors. Adapting NER systems across domains is challenging (Goyal et al., 2018), because of the significant differences in dictionaries for recognizing domain-specific NEs. Besides, most domains do not have enough quality data sets to build a mature language model. The lack of data prevents a "brutal-force" method by exhaustive dictionaries (Majumder et al., 2012). Specifically, the construction area has unique NEs that are found in different types of documents. Many of the documents have a weak structure and contain colloquial words, which further increases the difficulty.

Grammar factors. Most of the earlier NER studies focus on European languages (e.g., English, French, etc.), in which the developed NER models could not be used for Chinese documents for three reasons. First, Chinese has no capitalization for identifying proper noun NEs (e.g., locations, names, and organizations). Second, Chinese has no word inflection and derivation, which makes the recognition of the correct word part-of-speech (POS) (i.e., word classes) difficult. For example, the verb "protect" and noun "protection" share the same Chinese word "支护". As a result, recognizing the POS of "支护" from phrase "边坡支护 (slope protection)" is challenging. Third, Chinese has no space cues, thus making the segmentation of Chinese sentences into words difficult.

Entity type factors. Accurately defining the scope of NEs is essential to adapt to the needs of construction management. Specifically, it requires a construction NER framework that covers the NEs for different construction management tasks with clearly defined boundaries between different NE types. The lack of such a framework may cause inconsistent annotation and further affect the learning of language models. For example, without a predefined definition or specification, people may have different answers about whether "concrete formwork" should be classified as a "building material" or a "construction tool", or whether "cantilever beam stirrup" should be considered as a single entity with two nested elements or two entities. A potential solution is to build a structured hierarchy that defines most NE types and rigid designators frequently appearing in the domain documents (Li et al., 2016).

In summary, two key challenges arise. First, building a NER corpus in Chinese from scratch is difficult. A high-quality Chinese corpus in the construction domain requires a well-designed strategy to define target NEs for construction management and to ensure the validity of NE annotation. It can significantly facilitate later works, but needs an extensive initial effort to build. Second, it lacks knowledge about effective features that may critically affect the performance of a learning-based NER model. A learning-based model intrinsically has the potential to overcome many problems caused by grammatical factors, such as varied description for the same object, colloquial words, and the lack of word inflection or derivation. Thus, choosing effective features to ensure the performance of a learning-based NER model is essential, given the limited training data in the construction domain.

This study presents a NER method based on conditional random field (CRF) for Chinese construction documents, aiming at addressing the above-mentioned problems. The presented work includes a corpus design strategy, a CRF-based NER model with feature selection, and detailed performance analysis. The performance analysis reveals findings on error causes, feature effects, and impact of data volume to illustrate future research directions.

2 Related work

2.1 Natural language processing (NLP) in construction

NLP has been applied in many construction engineering and management fields, tackling problems such as information retrieval, text classification, automated compliance checking, and knowledge mining. The vector space model and the statistical language model are the two major types of model used in information retrieval (Singhal, 2001; Lv and El-Gohary, 2016a; Zou et al., 2017). The vector space model represents a query and a document as two vectors of term. The similarities between query vectors and document vectors are calculated and ranked to retrieve the most relevant documents to the query (Zou et al., 2017). The statistical language model considers documents as a sample that represents the distribution of words from a language model and ranked according to their relevance to a query (Singhal, 2001; Lv and El-Gohary, 2016a). Representative examples of application include the retrieval of environmental information in transportation project documents (Lv and El-Gohary, 2016a) and construction dispute (Zou et al., 2017).

Text classification has the potential to improve the efficiency of construction document management. Promising methods proposed include vector space model, ontology-based models (Al Qady and Kandil, 2010), and latent semantic analysis (Al Qady and Kandil, 2013). For example, Al Qady and Kandil (2015) and Caldas and Soibelman (2002) employed the vector space model to automatically organize construction project documents. Zhou and El-Gohary (2016) used ontology to improve the classification of codes in environmental regulatory textual documents. Al Qady and Kandil (2013) utilized latent semantic analysis to classify project documents on the basis of document discourse.

Automated compliance checking is another direction that has been extensively studied in the construction domain and implemented in designing, safety

management, and underground utility management (Zhang and El-Gohary, 2015; 2016; Xu and Cai, 2020). In automated compliance checking, information from regulations, specifications, or building codes are extracted in a computer-comprehensible format to support downstream comparison with designing models or geospatial data. The studies evidenced NLP's strong capability to automatically extract critical information. Xu and Cai (2020) employed a rule-based NLP method to translate unstructured textual spatial configurations into unified spatial rules to facilitate automated compliance checking of underground utilities. In their study, a specification language model was built, and a series of rules based on POS, gazetteer, and chunking were defined to process text into structured information. Zhang and El-Gohary (2015; 2016) proposed similar approaches to extract formalized information from various construction regulatory documents and have achieved high precision and recall scores.

NLP also shows a great potential to mine latent knowledge from a corpus. For example, Kwayu et al. (2020) analyzed hazardous actions on the basis of car crash narratives by using a NLP-based method. In their research, *n*-grams were extracted and filtered to understand how police officers assign hazardous actions to at-fault drivers. Chen and Luo (2019) conducted an analysis on scientific and engineering research on the basis of the abstracts of relevant literature. Noun phrases were extracted by using pattern-matching-based method and then filtered according to Shannon Entropy. Social network analysis was conducted on the remaining phrases to explore the latent knowledge.

2.2 Named entity recognition (NER) in construction

An essential finding by many existing studies is that the incorporation of NEs may significantly enhance the performance of NLP applications. In Lv and El-Gohary (2016a)'s research where a semantic annotation method was proposed to facilitate information retrieval in the transportation project environmental review domain, an epistemology containing domain concept terms was built and integrated into document representation. Their later study (Lv and El-Gohary, 2016b) further considered the semantic relatedness between the domain concepts and achieved promising improvement. Le et al. (2018) also argued that computers' understanding of technical terms/keywords for information retrieval is important. Hahm et al. (2015) designed an ontology-based method to help retrieve existing engineering documents accurately, in which domain terms' relationship was considered to calculate the relevance between queries and documents. In Zhou and El-Gohary (2016)'s text classification work for construction regulatory documents, words were identified and classified according to a proposed ontology to enhance the result.

Two NER methods are commonly used: Rule-based and learning-based methods. A rule-based method often relies on linguistic rules, statistical information, and the use of information lists such as gazetteers or taxonomies. A learning-based method aims at establishing a model and optimizing relevant parameters on the basis of labeled corpus.

2.2.1 Rule-based NER method

A typical rule-based NER method directly matches words/phrases in a sentence with a manually created taxonomy (Hahm et al., 2015; Lv and El-Gohary, 2016b; Lee et al., 2019). The precision of such a method is high, but building a taxonomy that can cover all possible cases is challenging, considering the diversity of language expression.

Some studies identify words/phrases on the basis of statistical information such as frequency (Hahm et al., 2015), term frequency–inverse document frequency (TF–IDF) (Sun et al., 2020), and C-value (Frantzi et al., 2000). Lv and El-Gohary (2016a) proposed a shallow semantic annotation algorithm utilizing statistical information and WordNet, a lexicon dictionary, to mine concept terms for retrieving transportation project environmental review. However, such methods cannot assign words/phrases with specific categories effectively. Without exhaustive rules, the result can be low in accuracy and needs to be manually checked (Zhang et al., 2019).

The method combining a manually made taxonomy with a set of rules can identify semantic information elements in building codes with good performance (Li et al., 2016; Tixier et al., 2016; Zhang and El-Gohary, 2016; Xu and Cai, 2020). However, the algorithm highly relies on the coverage of the lexicon, and additional expert effort is required to update the extraction rules and relevant ontology/taxonomy for new types of text. It requires a certain level of human expertise in domain-related knowledge, language knowledge, and at least basic programming skills. Moreover, it cannot be transferred across domains. The lack of portability further increases the cost to build and maintain a rule-based system dedicated to a specific domain.

By nature, a rule-based approach rarely makes mistakes within the coverage of the rules, but it is not capable of handling any situation outside the scope. Thus, it usually has high precision but low recall (it may miss many entities) (Jauregi Unanue et al., 2017) and is ineffective in the presence of word variations and abbreviations. For this study, word variations and abbreviations may frequently occur in Chinese construction documents such as daily reports, meeting minutes, and even specifications (see examples in Table 1).

2.2.2 Learning-based NER method

In comparison to rule-based method, learning-based

Table 1 Examples of word variation and/or abbreviation

Word	Variation/Abbreviation
外悬挑梁 (cantilever beam)	悬挑外梁
加气混凝土砌块 (aerated concrete block)	混凝土加气块, 加气块
水泥粉喷搅拌桩 (cement powder spray pile)	粉喷搅拌桩, 粉喷桩

system mainly uses machine learning techniques to identify language patterns within texts (Saha et al., 2012). It is intrinsically robust to variations (Jauregi Unanue et al., 2017), because it does not rely on any preset dictionary or vocabulary list. A typical procedure of a learning-based NER system includes building labeled training data with positive and negative examples, performing feature engineering on the basis of examples, and training a NER language model that can identify NEs by consuming features.

Despite the achievements in many areas, the learning-based NER systems have not drawn enough attention from researchers in the construction domain (Fan et al., 2015). Designing a dedicated NER model to process construction documents is essential, because a learning-based NER model will have a significant drop in performance (20% to 40% of precision and recall) when it is transferred from other domains (Poibeau and Kosseim, 2001). In recent studies, Liu and El-Gohary (2017) proposed a semi-supervised CRF model for information extraction based on an ontology and achieved an F-1 measure of 90.7%. Despite the remarkable success, it requires careful and tedious manual work to build a reliable ontology. Though the methods have shown promising results, the process can be complex and requires professional experience. Exploring an easily used method for NER is important to facilitate the multiple NLP tasks in the construction domain.

A learning-based system always requires a large volume of labeled data for training. Although numerous corpora exist, such as The Blog Authorship Corpus and Amazon reviews (Leskovec, 2013), no corpus for Chinese construction document-related NLP tasks exists. The lack of corpora becomes a gap that must be bridged. Besides, only a few studies have been conducted on Chinese NER in the construction domain. With the booming of infrastructure construction in China, a significant number of ongoing projects are using Chinese as the main communication language, causing a huge demand for an efficient Chinese NER system to enhance construction automation and information management.

2.3 Conditional random field (CRF) model

The NER problem can be formulated as a sequence labeling problem: Given a word sequence $W = [w_1, w_2, ..., w_n]$, where w_i represents the ith word in W, a NER task labels each word by an element in the label set $L = \{B, I, O\}$. The CRF model is popular in addressing sequence labeling problems. A typical CRF model can be expressed as:

$$P(l \mid w) = \frac{1}{Z(w)} \exp\left(\sum_{i,k} \lambda_k t_k(l_{i-1}, l_i, w, i) + \sum_{i,j} \mu_j s_j(l_i, w, i)\right), \quad (1)$$

where

$$Z(w) = \sum_{l} \exp\left(\sum_{i,k} \lambda_k t_k(l_{i-1}, l_i, w, i) + \sum_{i,j} \mu_j s_j(l_i, w, i)\right). \quad (2)$$

In the formula, l represents a possible labeling sequence, w is the input word sequence, λ_k and μ_j represent the corresponding weights, t_k is the transition feature function, and s_j is the state feature function. The value of t_k or s_j is 1 when the feature conditions are met or 0 otherwise. $Z(w)$ is the normalization factor, and the summation is performed on all possible output sequences. $P(l \mid w)$ represents the conditional probability predicted for the output sequence l given the word sequence w. The labeling sequence $l_{\max}$ that makes $P(l \mid w)$ largest is the output of a CRF model for an input w.

The feature functions need to be designed and serve as a foundation for model training and evaluation. Model training is a process that adjusts the weights λ_k and μ_j according to the training data in order for the model to correctly predict the labeled sequence of unknown sentences. The model evaluation work assesses the performance of the trained model on test data to verify the robustness (Lafferty et al., 2001).

3 CRF-based NER for Chinese construction documents

This section introduces the CRF-based NER model for Chinese construction documents, including the corpus design strategy and the CRF modeling procedure with feature selection.

3.1 Corpus design strategy

The corpus in this study is designed in three steps, that is, task definition to identify target NEs for construction management tasks, work segmentation to prepare annotating tokens, and NE annotation with a specification to ensure annotating quality. Figure 1 illustrates the pipeline.

3.1.1 Task definition

A learning-based NER model requires a large volume of

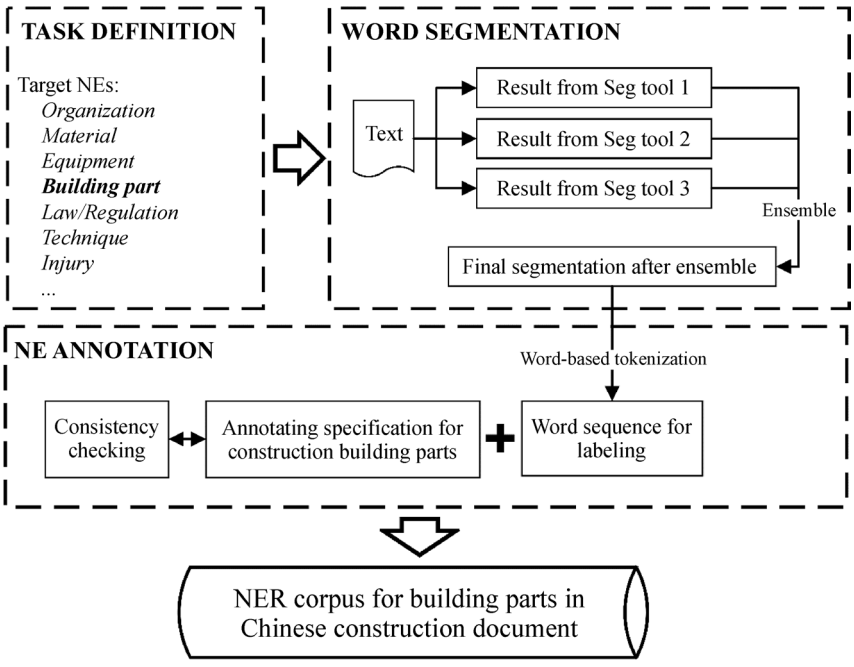

Fig. 1 Corpus design pipeline.

data to ensure its accuracy, and the required amount of training data will rise dramatically as the number of target NE types increases. A clear definition of NER tasks can help improve the performance with limited data. For example, the study of content analysis for construction injury reports defined body parts, energy sources, and injuries as the target NEs (Tixier et al., 2016); whereas the study of building code extraction considered materials and building components as the target NEs (Zhou and El-Gohary, 2016). In light of prior research and discussion with construction professionals, we identified the typical NER tasks, the target NEs, and the types of documents related to construction management as shown in Table 2.

This study focuses on building parts as the target. A building part is defined as a building element (e.g., beam and column) with adjective information such as location (e.g., site district and site gate) and material (e.g., concrete and steel). It usually determines the object of discussion or argument in a construction document. The identification of building parts can facilitate many NLP tasks including document classification, text summarization, and information extraction. In addition, we maintain the generality to ensure that the method developed here can be applied to identify other NEs.

A large amount of building parts in Chinese construction documents contain nested entities, which significantly increases the difficulty for their identification. Addressing the problem requires an accurate Chinese word segmentation method to support a word-based annotation process and a well-designed annotating specification to control potential inconsistency issues. The rest of this section provides the details of an ensemble method to improve Chinese word segmentation and a formalized annotating specification for building parts in Chinese construction documents.

3.1.2 Word segmentation

Word segmentation in Chinese plays a critical role and serves as a foundation to support word-based tokenization. Word segmentation splits sentences into word sequences.

Table 2 NER tasks, target NEs, and associated construction documents

Task	Target NEs	Construction document
Identification of responsibility and legal issues	Organization Party Law/Regulation	Contract Bidding document correspondence Other formal communication
Cost analysis	Material Equipment Building parts	Progress report Project quota
Progress analysis	Material Equipment Building parts	Construction plan Progress report Daily log Meeting minutes
Quality analysis	Material Building parts Construction techniques	Construction plan Quality report Daily log Meeting minutes
Safety analysis	Equipment Building parts Date Body parts Injury types	Construction plan Safety report Daily log Meeting minutes

Some existing segmentation tools for Chinese have achieved acceptable accuracy in general corpora, but their accuracy dramatically decreases when processing domain texts. To improve the accuracy, we adopt a two-step method: 1) build a domain dictionary with 13612 words collected from *An Encyclopedia of Architecture and Civil Engineering of China*, and integrate it into three widely-used word segmentation tools, namely, Language Technology Platform (LTP) (Che et al., 2010), Jieba (available at github.com/fxsjy/jieba), and THU Lexical Analyzer for Chinese (THULAC) (Li and Sun, 2009); and 2) design the ensemble method that fuses the results of the three segmentation tools. The procedure of the ensemble method is shown in Table 3. Figure 2 demonstrates an example in which the sentence "第三层高低跨转换梁长度不足。(The length of the stepped force-transferring beam on the third floor is not sufficient.)" is segmented by three different NLP tools with the help of a domain dictionary. In each segmentation result, the characters of each token are labeled as "F" or "E" according to whether they are the "first" or "ending" character of the token. For example, the tag of character "层" is "F" in the segmentation result of LTP and "E" in that of Jieba. The final tag of "层" is "F", for the number of tag "F" for "层" is larger than that of "E" among the three results. The final segmentation result is then transferred according to the "F–E" sequence of characters.

We labeled 150 sentences from the daily reports of a construction project by using the labeling rules developed by Yu et al. (2018) and comparing the ensemble method with the three individual models. As shown in Table 4, the ensemble method is approximately 5% more accurate than the others.

3.1.3 Named entity (NE) annotation

This step annotates the NEs according to the results of word segmentation, with each word (instead of a character) considered as a token. This study adopts the "*BIO*" format (Table 5) as the segmentation representation.

Annotating inconsistency is a major challenge, where a token may be assigned with different tags by different annotators. We design an annotating specification according to the "Model-Annotate-Model-Annotate" cycle

Table 3 Procedures of the ensemble method

// *Prepare initial tag sequences before fusion*:

[1] For each sentence $s = [c_1, c_2, ..., c_j, ..., c_N]$, where c_j represents the jth character in s, record the results from the three segmentation tools as R_1, R_2, and R_3

[2] Mark the first character of each word by tag "F" and the rest by tag "E" in R_1, R_2, and R_3

[3] Let $T_i = [t_{i1}, t_{i2}, ..., t_{ij}, ..., t_{iN}]$ represents the tag sequence of R_i, where t_{ij} represents the jth tag of the jth character in R_i

// *Form the final tag sequence by fusion*:

[4] For each character c_j in s, a list $T_j = [t_{1j}, t_{2j}, t_{3j}]$ exists

[5] Count the number of tag "F" and tag "E" in T_j as N_{Fj} and N_{Ej}, respectively

[6] Assign "F" or "E" as the final tag of character c_j on the basis of $\max(N_{Fj}, N_{Ej})$ and form a final tag sequence T_{Final}

// *Transform the final tag sequence into segmented tokens as the fusion result*:

[7] Scan from left to right; mark an F before an F as an individual token

[8] Mark an F before an E as the beginning of a token; mark an E before an F as the end of a token; search back for the nearest beginning of a token; and combine the beginning–end pair together with every character in between as a token

Fig. 2 Illustration of the ensemble method.

Table 4 Accuracy comparison

Model	LTP	Jieba	THULAC	Ensemble
Accuracy	0.905	0.914	0.918	0.963

Table 5 Tag representation

Tag location	Beginning	Inside	Outside
Tag representation	*B*	*I*	*O*

(Pustejovsky and Stubbs, 2012) to control potential inconsistency issues. The main consideration of the design is to avoid vague boundaries and maximize the potentials to facilitate downstream tasks, for example, better representation of the target of a discussion or argument to facilitate text classification. The fundamental rules of the specification are as follows:

(1) All nested elements are categorized into three types that, respectively, represent a location, a building component, and a building material (Table 6);

Table 6 Nested NE element types

Types	Example (in Chinese)	Example (in English)
Location	顶层, 一区	top floor, zone one
Building components	梁, 墙, 10#塔吊	beam, wall, 10# tower crane
Building material	混凝土, 砖, 钢	concrete, brick, steel

(2) For a NE string with nested elements of "location", split it after each "location" elements. For example, in the string "A区5层悬挑梁底部箍筋", "区 (area)", "层 (floor)", and "底部 (bottom)" are of the types of "location"; hence, the string will be split into "A区 (area A)", "5层 (5th floor)", "悬挑梁底部 (cantilever beam bottom)", and "箍筋 (stirrup)".

(3) For a NE string without a nested element of "location", take the entire string as a single NE. For example, though the string "悬挑梁箍筋 (cantilever beam stirrup)" contains two tokens ("悬挑梁 (cantilever beam)" and "箍筋 (stirrup)"), it will be treated as a single NE.

A set of supervisory reports containing a total of 759 sentences were collected from two construction projects as the experimental data. Three annotators with engineering knowledge were invited to annotate individually the data according to the specification. To quantify the annotation consistency, the Kappa score was used to measure the pairwise agreement of annotations. The Kappa value is calculated by $\kappa = \frac{\Pr(a) - \Pr(e)}{1 - \Pr(e)}$, where $\Pr(a)$ represents the proportion of words that are labeled the same by the annotators to the total number of words and $\Pr(e)$ represents the probability that a word is labeled with the same tag by the annotators (Al Qady and Kandil, 2010). In general, the larger the value of κ is, the more consistent the annotations are.

The Kappa values were calculated between the tag sets by each pair of annotators as shown in Table 7. The average value is 0.93, indicating high consistency (Pustejovsky and Stubbs, 2012). The result of annotator A was selected as the final corpus, because its average Kappa score was the highest. The corpus contains a total of 15213 words (24839 characters) and is written in CSV format. The first column lists the words, and the second column lists the corresponding labels. The corpus and relevant information are available at github.com/isotrforever/NER-corpus-for-construction.

Table 7 Annotation consistency matrix

Annotator	A	B	C	Average Kappa
A	–	92.8%	93.9%	93.4%
B	92.8%	–	92.3%	92.6%
C	93.9%	92.3%	–	93.1%
Average	93.4%	92.6%	93.1%	93.0%

3.2 CRF-based NER model

The performance of a CRF model depends highly on the features selected. This study defines nine transformation features (TFs) and nine classes of state features (SFs). The SFs include POS state features (POSFs), four classes of word state features (WFs), and four classes of character state features (CFs). The TFs describe a word's position, whereas the SFs describe a word's or a character's states within its context.

3.2.1 Transformation features (TFs)

The TFs are usually adopted as important features in a CRF model. The TF functions can be presented as:

$$TF_{L1,L2}(l_{i-1},\ l_i,\ w,\ i) = \begin{cases} 1, & l_{i-1} = L1,\ l_i = L2 \\ 0, & \text{otherwise} \end{cases},$$

$$L1,\ L2 \in \{B,\ I,\ O\}, \tag{3}$$

where $L1$, $L2$ refers to two adjacent words/characters, with $L1$ being the former and $L2$ being the latter; l_{i-1} and l_i represent the tag of the $(i-1)$th word and the ith word, respectively; w is the word sequence; and i is the word's order in the sentence. Given that $L1$ and $L2$ have three possible values — B, I, and O — the total number of TFs is 9.

3.2.2 State features (SFs) — POS

Part-of-speech information represents the grammatical roles of a word. Words with the same POS tag have similar grammatical properties. The POS information is

obtained using the LTP Chinese NLP tool in this study. Each token is assigned by a POS tag in the LTP tag set. The definitions of the tags can be found on its website (available at ltp.ai/docs/appendix.html#id2). The POSF functions are presented as:

$$POSF_{L,POS(l_i,w,i)} = \begin{cases} 1, & l_i = L,\ pos(w_i) = pos \\ 0, & \text{otherwise} \end{cases}, \quad (4)$$

where w_i is the ith word in word sequence w; pos represents one of the POS tags in the LTP tag set; $pos(w_i)$ is the POS feature of word w_i; and L belongs to the label set $\{B, I, O\}$. The total number of POSF functions is equal to the size of POS tag set multiplied by the size of the label set.

3.2.3 State features (SFs) — Word state and character state

The word and character state features are related to four word types and four character types (see examples in Table 8). Intuitively, the word state and character state of a word can indicate the possibility of the word or its neighbors being an element of a NE. For example, the succeeding word of the word "的 (of)" is the beginning word of a NE in many cases; thus, the word "的 (of)" can be a left-hand indicator. The suffix of a word also indicates the role of the word in a sentence. For example, if the character "筋 (bar)" appears in the last position of a word, then the word may be a building part with a high probability. To capture such information, four word types and four character types are designed.

The four word types are:

- Kernel: The last word of a NE;
- Modifier: A word other than the "kernel" in a NE;
- Left-hand indicator: The word to the left of a NE;
- Right-hand indicator: The word to the right of a NE.

The four character types are:

- Single kernel suffix: The last character of a kernel;
- Double kernel suffix: The last two characters of a kernel;
- Single modifier suffix: The last character of a modifier;
- Double modifier suffix: The last two characters of a modifier.

A word of a specific type is confirmed in the corresponding state only when it appears frequently enough, that is, the time that it appears in the text is more than a threshold. For example, if the word "beam" appears as a "kernel" 10 times in the training corpus and the threshold of the kernel state is set as 8, then the word "beam" is confirmed as a "kernel". The WF function can be presented as:

$$WF_{L,\text{kernel}}(l_i,\ w,\ i) = \begin{cases} 1, & l_i = L,\ f_{\text{kernel}}(w_i) > TH(\text{kernel}) \\ 0, & \text{otherwise} \end{cases}, \quad (5)$$

where $f_{\text{kernel}}(w_i)$ is the frequency that w_i shows in the text as the type of "kernel", and $TH(\text{kernel})$ is the threshold set for the WF of "kernel". A WF's threshold plays a critical role in determining whether a word is in the corresponding state. A low threshold may affect the accuracy of recognition, whereas a high one may affect the efficiency. It is usually initialized with an empirical number and finalized through a trial-and-error process. The other three WF functions are:

$$WF_{L,\text{modifier}}(l_i,\ w,\ i) = \begin{cases} 1, & l_i = L,\ f_{\text{modifier}}(w_i) > TH(\text{modifier}) \\ 0, & \text{otherwise} \end{cases}, \quad (6)$$

$$WF_{L,\text{left\_indicator}}(l_i,\ w,\ i) = \begin{cases} 1, & l_i = L,\ f_{\text{left\_indicator}}(w_i) > TH(\text{left\_indicator}) \\ 0, & \text{otherwise} \end{cases}, \quad (7)$$

Table 8 Types of words/characters in a sentence

Word	Tag	Word type	Character type
请	*O*		
施工	*O*		
单位	*O*		
尽快	*O*		
上报	*O*	Left-hand indicator	
人工	*B*	Modifier	"工" is a single modifier suffix and "人工" is a double modifier suffix
挖孔	*I*	Modifier	"孔" is a single modifier suffix and "挖孔" is a double modifier suffix
桩	*I*	Kernel	"桩" is a kernel suffix
和	*O*	Left-hand indicator and right-hand indicator	
土钉	*B*	Modifier	"钉" is a single modifier suffix and "土钉" is a double modifier suffix
墙	*I*	Modifier	"墙" is a modifier suffix
锚杆	*I*	Kernel	"杆" is a single kernel suffix and "锚杆" is a double kernel suffix
的	*O*	Right-hand indicator	
变更	*O*		
费用	*O*		
。	*O*		

$$WF_{L,\text{right\_indicator}}(l_i,\ w,\ i) = \begin{cases} 1, & l_i = L,\ f_{\text{right\_indicator}}(w_i) > TH(\text{right\_indicator}) \\ 0, & \text{otherwise} \end{cases}. \tag{8}$$

Similarly, the CF functions can be presented as:

$$CF_{L,\text{s\_kernel\_suffix}}(l_i,\ w,\ i) = \begin{cases} 1, & l_i = L,\ f_{\text{s\_kernel\_suffix}}(w_i) > TH(\text{s\_kernel\_suffix}) \\ 0, & \text{otherwise} \end{cases}, \tag{9}$$

$$CF_{L,\text{d\_kernel\_suffix}}(l_i,\ w,\ i) = \begin{cases} 1, & l_i = L,\ f_{\text{d\_kernel\_suffix}}(w_i) > TH(\text{d\_kernel\_suffix}) \\ 0, & \text{otherwise} \end{cases}, \tag{10}$$

$$CF_{L,\text{s\_modifier\_suffix}}(l_i,\ w,\ i) = \begin{cases} 1,\ l_i = L,\ f_{\text{s\_modifier\_suffix}}(w_i) > TH(\text{s\_modifier\_suffix}) \\ 0,\ \text{otherwise} \end{cases}, \tag{11}$$

$$CF_{L,\text{d\_modifier\_suffix}}(l_i,\ w,\ i) = \begin{cases} 1,\ l_i = L,\ f_{\text{d\_modifier\_suffix}}(w_i) > TH(\text{d\_modifier\_suffix}) \\ 0,\ \text{otherwise} \end{cases}. \tag{12}$$

where $f_{\text{s\_kernel\_suffix}}(w_i)$ represents the frequency of the last character of the word w_i shown as kernel suffix, and so does $f_{\text{s\_modifier\_suffix}}(w_i)$; and $f_{\text{d\_kernel\_suffix}}(w_i)$ represents the frequency of the last two characters of the word w_i shown as kernel suffix (if w_i contains only one character, then the value is 0), and so does $f_{\text{d\_modifier\_suffix}}(w_i)$.

Once the features are determined, the model training process will adjust the weights of the feature functions. We take approximately 80% of the annotated data as the training data and the remaining 20% as the test data. The weight adjustment task then employs an iterative process to find parameters λ_k and μ_j that maximize the log-likelihood of the training data (Lafferty et al., 2001). During the model evaluation process, we compare the predicted tag sequences on the test data with the manually annotated gold standard. The precision, recall, and F1-measure are calculated as:

$$precision_L = \frac{S_L \cap S_L'}{S_L'}, \tag{13}$$

$$recall_L = \frac{S_L \cap S_L'}{S_L}, \tag{14}$$

$$F1\text{-}measure_L = \frac{2 \times precision_L \times recall_L}{precision_L + recall_L}, \tag{15}$$

where S_L denotes a set of words annotated as label L in the test data, and S_L' denotes a set of words annotated as label L by the algorithm.

4 Results analysis

The corpus mentioned in Section 3.1 is randomly divided into training data and test data in a proportion of 4:1. The training data consists of 607 sentences, whereas the test data consists of 152 sentences. The Sklearn-crfsuite package (available at sklearn-crfsuite.readthedocs.io/en/latest/) is used as the CRF modeling and training tool. The iteration number is 600; the optimization method is set as gradient descent; and the coefficient for $L1$ and $L2$ regularization are both set as 0.1. For a given corpus, as explained in Section 3.2, the model's accuracy can be affected by the thresholds. Table 9 shows the optimized thresholds for all word/character state features after comparing the performances of multiple attempts.

Table 9 Selected thresholds of statistical feature

Threshold name	Value
TH(kernel)	3
TH(modifier)	
TH(left_indicator)	1
TH(right_indicator)	
TH(s_kernel_suffix)	10
TH(s_modifier_suffix)	
TH(d_kernel_suffix)	
TH(d_modifier_suffix)	

4.1 Model performance

Table 10 presents the performance of the model on test data. The F-1 scores of tags *B*, *I*, and *O* are 0.812, 0.853, and 0.972, respectively. Figure 3 illustrates some exemplary results. The recognized NEs are marked blue with suffixes.

The performance of the introduced model is compared with that of Bi-LSTM-CRF (Huang et al., 2015) and BERT-Bi-LSTM-CRF (Devlin et al., 2018) using the same

Table 10 Performance of the CRF model

Tags	Precision	Recall	F1-measure
B	0.835	0.790	0.812
I	0.892	0.816	0.853
O	0.954	0.990	0.972
Average			0.879

training and testing data. Bi-LSTM-CRF and BERT-Bi-LSTM-CRF are widely used neural network models for NER in the computer science domain, and their good performance have been evidenced in many applications (Dai et al., 2019; Luo et al., 2018). For the Bi-LSTM-CRF model, the learning rate is 0.01, the batch size is 20, and the number of epochs is 30. For the BERT-Bi-LSTM-CRF model, the sequence length is 256, the learning rate is 2×10^{-5}, the batch size is 8, and the number of epochs is 30. Both models are trained on the computer with 3.70 GHz's Inter(R) Core(TM), 64G RAM and NVIDIA GeForce RTX 2080 Ti with 11048 MB VRAM. Table 11 lists the F-1 scores of the three models, and our model reaches the highest F-1 score of 0.879.

Table 11 Performance comparison

Model	Introduced model	Bi-LSTM-CRF	BERT-Bi-LSTM-CRF
F1-measure	0.879	0.813	0.827

A detailed investigation was conducted to identify the error types and potential causes. Most of the errors occur on the identification of nested entities. For example, the phrase "L6 层 钢筋 机械 连接 (L6 layer reinforcement mechanical connection)" is labeled as "*B I B I I*" by the annotator, whereas the model predicts it as "*B I I B I*". This type of error may lead to a mismatch between the NEs and the predefined specification. Another example is that in the phrase "排烟风管半成品加工不规范 (non-standard processing of semi-finished smoke exhaust ducts)", "风管 (duct)" and "半成品 (semi-finished)" were identified, whereas the word "排烟 (smoke exhaust)" is missing. A possible cause is related to the grammar factor that the Chinese language has no inflection or derivation. When a word in a "verb–noun" form (e.g., "排烟") is nested in another entity, the model has difficulty recognizing correctly. Despite the speculation, the actual cause and solution require future study.

4.2 Effects of POSF on performance

POS has been adopted as important features in many previous studies. To further investigate how POS affects performance, we compared the F1-measure results of different feature combinations. As shown in Table 12, the inclusion of POSF improved the performance in general, but the margin of improvement is decreasing as more features are considered. Experiment No. 5 without POSF showed a similar performance with Experiment No. 6, and the prediction results of "*I*" and "*O*" are even better. The reason may be that the addition of POS features introduces more parameters to be optimized during training. When sufficient domain knowledge features are provided, the marginal information gain brought by the POS features is not enough to overcome the negative impact of the increased parameters. This finding leads to a promising future research direction of testing with a simplified POS setting, that is, noun vs. others. Most POS features identified in Table 13 are nouns (b, nd, nh, nz, and nt) or directly related to noun (a, p, and q). Moreover, a POS framework usually has much more tags than the ones listed here. For instance, the National Standard 863 POS Tagging Set for Chinese has 28 types of tag, and the Penn Treebank for English has 36 types. Simplifying the tag set may reduce the dimension without losing critical information and improve the accuracy with limited data. Some existing studies have demonstrated the potential, such as in the research by Tixier et al. (2016), where the injury precursor extraction system has reached high accuracy without the POS features.

监理组在现场巡视发现[@负二层#Component*][@一区#Component*][@墙柱钢筋#Component*]未按拆撑方案搭设[@防护#Component*]，请施工方严格按照拆撑方案施工；

监理组现场巡视时发现你单位[@连廊#Component*]灌浆拆模后空鼓露筋现象严重，现要求你单位编制可靠的修补方案，进行修补整改到位；

监理在现场巡视中，发现[@一区#Component*][@负二层#Component*][@排架#Component*]搭设过程中[@立杆底部#Component*]未按[@模板#Component*]方案要求设置400*100*50[@木垫块#Component*]，请施工方严格按照方案要求施工，对[@现场立杆#Component*]未设置[@垫木#Component*]的地方及时整改，整改完成报监理验收。

监理组检查发现[@B3层#Component*][@地库外墙#Component*]局部出现通透裂缝，为确保工程结构质量，现要求施工方就此类质量缺陷提出可靠有效的处理方案，报监理、建设方同意后，及时按照审核同意的处理方案认真整改处理！

监理组检查发现[@塔楼区域#Component*][@B3层#Component*][@地库顶板#Component*]多处出现通透裂缝，为确保工程[@结构#Component*]质量，现要求施工方就此类质量缺陷提出可靠有效的处理方案，报监理、建设方同意后，及时按照审核同意的处理方案认真整改处理！

Fig. 3 Visualization of part of the results.

Table 12 F1-measures of different feature combinations

No.	Features	F1-measure of tag *B*	F1-measure of tag *I*	F1-measure of tag *O*	Average
1	TF, CF	0.653	0.766	0.944	0.788
2	TF, CF, POSF	0.742	0.809	0.957	0.836
3	TF, WF	0.745	0.814	0.975	0.845
4	TF, WF, POSF	0.781	0.839	0.977	0.866
5	TF, WF, CF	0.798	0.859*	0.978*	0.878
6	TF, WF, CF, POSF	0.812*	0.853	0.972	0.879*

Note: * marks the largest number in each column.

Table 13 Parameters of the SFs (top six)

Tag location	Weight	Feature
$L = B$	1.968	$S_{L=B,\ pos=b}$
	1.472	$S_{L=B,\ pos=nh}$
	1.207	$S_{L=B,\ \text{s\_modifier\_suffix}}$
	1.201	$S_{L=B,\ pos=ws}$
	1.191	$S_{L=B,\ pos=n}$
	−1.635	$S_{L=B,\ pos=q}$
$L = I$	3.056	$S_{L=I,\ pos=wp}$
	1.601	$S_{L=I,\ \text{kernel}}$
	1.124	$S_{L=I,\ pos=nz}$
	−1.296	$S_{L=I,\ pos=b}$
	−1.460	$S_{L=I,\ \text{right\_indicator}}$
	−1.574	$S_{L=I,\ pos=a}$
$L = O$	2.841	$S_{L=O,\ pos=p}$
	2.064	$S_{L=O,\ pos=wp}$
	1.964	$S_{L=O,\ pos=nt}$
	−2.168	$S_{L=O,\ \text{modifier}}$
	−2.454	$S_{L=O,\ pos=nh}$
	−3.077	$S_{L=O,\ \text{kernel}}$

Notes: Definition of the POS tags: a – adjective, b – other noun-modifier, p – preposition, q – quantity, wp – punctuation, ws – foreign words, n – general noun, nd – direction noun, nh – person's name, nz – other proper noun, nt – temporal noun.

4.3 Impact of training data volume

To explore the impact of training data volume on model performance, we calculate the F1-measures with different scales of training data. Figure 4 demonstrates that the model performance is approaching an asymptotic level with limited data. It also indicates a space for improvement with more training data provided. Considering the scope of this study, the exploration with different levels of data amount and the best practice to maximize the usage of training data will be presented in future studies.

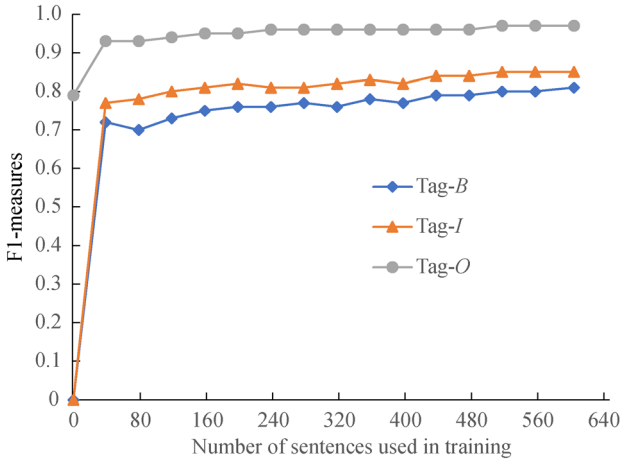

Fig. 4 F1-measures with different amounts of training data.

5 Conclusions

This study presents a CRF-based NER method to identify building parts from Chinese construction documents automatically. The proposed methodology achieves an F1-measure of 87.9%. Compared with the previous methods in the construction domain, it avoids the tedious lexicon-building and rule-making process. The proposed NER methodology has the potential to be transformed to recognize other types of NEs in different domains and can facilitate many downstream information processing tasks in the construction sector.

This study contributes to the body of knowledge in three aspects. First, it introduces a NER corpus design strategy for Chinese construction documents. The strategy includes a task definition framework, a Chinese word-based segmentation method, and a NE annotation specification. The resemble method during segmentation can be applied to general Chinese pre-process tasks in other domains. Second, it introduces a CRF-based NER model for Chinese construction documents with feature selection and detailed performance analysis. It can avoid the tedious rule-making

process and be conveniently adopted to facilitate many downstream NLP tasks. Third, the marginal performance improvement of including POS features will decrease as the model learns more domain knowledge. Simplifying or even abandoning the POS feature may facilitate the rule-based system designing process or feature engineering tasks.

As a fundamental method, the presented work can significantly facilitate constructing domain gazetteers and improve the robustness during NE matching. It provides several practical values potentially leading to the improvement of construction management efficiency. The corpus and its design strategy can serve as a starting point for more researchers and practitioners to contribute to the establishment of an open construction NLP data set. With more training data available, combined with the modeling process that avoids the effort of lexicon-building and rule-making, more practical applications can be developed with better accuracy and robustness. The automatic structuring of building codes, as an illustrative example that will benefit from an efficient NER method, can facilitate construction management by converting non-formatted building codes into computer-readable information to check the compliance of building models or construction processes. Several studies have used rule-based approaches to identify NEs and filled them to templates (Zhang and El-Gohary, 2015; 2016; Xu and Cai, 2020) for compliance checking in English. Automatic contract risk identification is another example. It refers to the task of detecting inappropriate descriptions/poisonous clauses in a contract for risk management. NER can support the task by identifying the semantic categories of NEs in a clause and matching them with predefined risk expression patterns, as shown in similar work by Lee et al. (2019).

The major limitation of the presented NER method is its reliance on well-annotated data. The scarcity of annotated data is and will be a challenge for NLP in construction management for a certain period of time. Exploring unsupervised or semi-supervised methods to make full use of unlabeled data is promising to ease such a problem. Another limitation is that the presented method only considers one NE category at a time. In practice, construction management tasks usually require the identification of multiple types of NE, for example, extracting a construction event by identifying the subject, the behavior, and the object. Further research effort is needed to design a method for multiple NE recognition at the same time.

References

Al Qady M, Kandil A (2010). Concept relation extraction from construction documents using natural language processing. Journal of Construction Engineering and Management, 136(3): 294–302

Al Qady M, Kandil A (2013). Document discourse for managing construction project documents. Journal of Computing in Civil Engineering, 27(5): 466–475

Al Qady M, Kandil A (2015). Automatic classification of project documents on the basis of text content. Journal of Computing in Civil Engineering, 29(3): 04014043

Caldas C H, Soibelman L (2002). Implementing automated methods for document classification in construction management information systems. In: Proceedings of the International Workshop on Information Technology in Civil Engineering. Washington, D.C.: ASCE, 194–210

Che W, Li Z, Liu T (2010). LTP: A Chinese language technology platform. In: Proceedings of the 23rd International Conference on Computational Linguistics: Demonstrations. Beijing: Association for Computational Linguistics, 13–16

Chen H, Luo X (2019). An automatic literature knowledge graph and reasoning network modeling framework based on ontology and natural language processing. Advanced Engineering Informatics, 42: 100959

Dai Z, Wang X, Ni P, Li Y, Li G, Bai X (2019). Named entity recognition using BERT BiLSTM CRF for Chinese electronic health records. In: Proceedings of 12th International Congress on Image and Signal Processing, Biomedical Engineering and Informatics (CISP-BMEI). Suzhou: IEEE, 1–5

Devlin J, Chang M W, Lee K, Toutanova K (2018). BERT: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint, arxiv:1810.04805

Fan H, Xue F, Li H (2015). Project-based as-needed information retrieval from unstructured AEC documents. Journal of Management Engineering, 31(1): A4014012

Frantzi K, Ananiadou S, Mima H (2000). Automatic recognition of multi-word terms: The C-value/NC-value method. International Journal on Digital Libraries, 3(2): 115–130

Gangadharan V, Gupta D (2020). Recognizing named entities in agriculture documents using LDA based topic modelling techniques. Procedia Computer Science, 171: 1337–1345

Goyal A, Gupta V, Kumar M (2018). Recent named entity recognition and classification techniques: A systematic review. Computer Science Review, 29: 21–43

Hahm G J, Lee J H, Suh H W (2015). Semantic relation based personalized ranking approach for engineering document retrieval. Advanced Engineering Informatics, 29(3): 366–379

Huang Z, Xu W, Yu K (2015). Bidirectional LSTM-CRF models for sequence tagging. arXiv preprint, arxiv:1508.01991

Jauregi Unanue I, Zare Borzeshi E, Piccardi M (2017). Recurrent neural networks with specialized word embeddings for health-domain named-entity recognition. Journal of Biomedical Informatics, 76: 102–109

Kwayu K M, Kwigizile V, Zhang J, Oh J S (2020). Semantic *n*-gram feature analysis and machine learning-based classification of drivers' hazardous actions at signal-controlled intersections. Journal of Computing in Civil Engineering, 34(4): 04020015

Lafferty J, McCallum A, Pereira F C N (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In: Proceedings of 18th International Conference on Machine Learning. San Francisco, CA: ACM, 282–289

Le T, Jeong H D, Gilbert S B, Chukharev-Hudilainen E (2018). Parsing

natural language queries for extracting data from large-scale geospatial transportation asset repositories. In: Proceedings of Construction Research Congress. New Orleans, LA: ASCE, 70–79

Lee J, Yi J S, Son J (2019). Development of automatic-extraction model of poisonous clauses in international construction contracts using rule-based NLP. Journal of Computing in Civil Engineering, 33(3): 04019003

Leskovec J (2013). Web data: Amazon reviews. Available at: snap.stanford.edu/data/web-Amazon.html

Li S, Cai H, Kamat V R (2016). Integrating natural language processing and spatial reasoning for utility compliance checking. Journal of Construction Engineering and Management, 142(12): 04016074

Li Z, Sun M (2009). Punctuation as implicit annotations for Chinese word segmentation. Computational Linguistics, 35(4): 505–512

Liu K, El-Gohary N (2017). Ontology-based semi-supervised conditional random fields for automated information extraction from bridge inspection reports. Automation in Construction, 81: 313–327

Liu X, Zhou M (2013). Two-stage NER for tweets with clustering. Information Processing & Management, 49(1): 264–273

Luo L, Yang Z, Yang P, Zhang Y, Wang L, Lin H, Wang J (2018). An attention-based BiLSTM-CRF approach to document-level chemical named entity recognition. Bioinformatics, 34(8): 1381–1388

Lv X, El-Gohary N M (2016a). Semantic annotation for supporting context-aware information retrieval in the transportation project environmental review domain. Journal of Computing in Civil Engineering, 30(6): 04016033

Lv X, El-Gohary N M (2016b). Enhanced context-based document relevance assessment and ranking for improved information retrieval to support environmental decision making. Advanced Engineering Informatics, 30(4): 737–750

Majumder M, Barman U, Prasad R, Saurabh K, Saha S K (2012). A novel technique for name identification from homeopathy diagnosis discussion forum. Procedia Technology, 6: 379–386

Manning C D, Schutze H (1999). Foundations of Statistical Natural Language Processing. Cambridge, MA: MIT Press

Poibeau T, Kosseim L (2001). Proper name extraction from non-journalistic texts. In: Proceedings of 11th Computational Linguistics in the Netherlands. Tilburg: Brill, 144–157

Pustejovsky J, Stubbs A (2012). Natural Language Annotation for Machine Learning: A Guide to Corpus-building for Applications. Sebastopol, CA: O'Reilly Media

Quimbaya A P, Múnera A S, Rivera R A G, Rodríguez J C D, Velandia O M M, Peña A A G, Labbé C (2016). Named entity recognition over electronic health records through a combined dictionary-based approach. Procedia Computer Science, 100: 55–61

Saha S K, Mitra P, Sarkar S (2012). A comparative study on feature reduction approaches in Hindi and Bengali named entity recognition. Knowledge-Based Systems, 27: 322–332

Singhal A (2001). Modern information retrieval: A brief overview. IEEE Data Engineering Bulletin, 24(4): 35–43

Sun J, Lei K, Cao L, Zhong B, Wei Y, Li J, Yang Z (2020). Text visualization for construction document information management. Automation in Construction, 111: 103048

Tixier A J P, Hallowell M R, Rajagopalan B, Bowman D (2016). Automated content analysis for construction safety: A natural language processing system to extract precursors and outcomes from unstructured injury reports. Automation in Construction, 62: 45–56

Xu X, Cai H (2020). Semantic approach to compliance checking of underground utilities. Automation in Construction, 109: 103006

Yu S, Duan H, Wu Y (2018). Corpus of multi-level processing for modern Chinese. Available at: opendata.pku.edu.cn/dataset.xhtml?persistentId=doi:10.18170/DVN/SEYRX5 (in Chinese)

Zhang F, Fleyeh H, Wang X, Lu M (2019). Construction site accident analysis using text mining and natural language processing techniques. Automation in Construction, 99: 238–248

Zhang J, El-Gohary N M (2015). Automated information transformation for automated regulatory compliance checking in construction. Journal of Computing in Civil Engineering, 29(4): B4015001

Zhang J, El-Gohary N M (2016). Semantic NLP-based information extraction from construction regulatory documents for automated compliance checking. Journal of Computing in Civil Engineering, 30(2): 04015014

Zhou P, El-Gohary N (2016). Ontology-based multilabel text classification of construction regulatory documents. Journal of Computing in Civil Engineering, 30(4): 04015058

Zhou P, El-Gohary N (2017). Ontology-based automated information extraction from building energy conservation codes. Automation in Construction, 74: 103–117

Zou Y, Kiviniemi A, Jones S W (2017). Retrieving similar cases for construction project risk management using Natural Language Processing techniques. Automation in Construction, 80: 66–76