https://doi.org/10.1016/j.autcon.2021.103608

A Transformer-based Deep Learning Model for Recognizing Communication-oriented Entities from Patents of ICT in Construction

Hengqin Wu [a], Geoffrey Qiping Shen [b], Xue Lin [c,*], Minglei Li [d], Clyde Zhengdao Li [a]

[a] College of Civil and Transportation Engineering, Shenzhen University, Shenzhen, China
[b] Department of Building and Real Estate, The Hong Kong Polytechnic University, Hong Kong, China.
[c] School of Government, Nanjing University, Nanjing, China
[d] Huawei Technologies Co., Ltd., Shenzhen, China

Abstract

The patents of information and communication technology (ICT) in construction are valuable sources of technological solutions to communication problems in the construction practice. However, it is often difficult for practitioners and stakeholders to identify the key communication functionalities from complicated expressions in the patent documents. Addressing such challenges, this study develops a deep learning model to enable automatic recognition of communication-oriented entities (CEs) from patent documents. The proposed model is structured based on the Transformer, consisting of feed-forward and self-attention neural networks to better recognize

Abbreviations: AI, artificial intelligence; BIM, Building Information Modeling; CE, communication-oriented entity; CEM, Construction Engineering and Management; CNN, convolutional neural networks; CM, communication models; CRF, conditional random fields; CS, communication subjects; FN, false negatives; FP, false positives; GRU, gated recurrent unit; ICT, information and communications technology; LSTM, long short-term memory; NLP, natural language processing; RFID, radio frequency identification; RNN, recurrent neural networks; TBNN, Transformer-based neural networks; TI, transferred information; TP, true positives; USPTO, United States Patent and Trademark Office; WoS, Web of Science ambiguous and unknown entities by utilizing contextual information. The validation results showed that the proposed model has superior performance in CE recognition than traditional recurrent neural networks (RNN)-based models, especially in recognizing ambiguous and unknown entities. Moreover, experimental results on some research literature and a real-life project report showed satisfactory performance of the model in CE recognition across different document types.

Keyword: Information and communications technology (ICT); Construction industry; Entity recognition; Deep learning; Transformer; Contextual information

1. Introduction

Information and communication technology (ICT) is an extensional concept, incorporating a wide range of technical approaches that mainly concentrate on communication functionalities [1]. The core benefit of ICT application in the construction industry is to enable and enhance communication, improving the coordination of data in the whole life cycle of construction projects [2,3]. Successful adoption of ICT relies on appropriate choices of technologies to enable desired communication functionalities according to specific objectives in construction practice [4,5]. In order to choose the right technologies for the confronting problems, practitioners and stakeholders need to fully comprehend the communication functionalities embedded in ICTs [2]. Patents are a common source for up-to-date technologies, from which 95% inventions can be found. The information of communication functionalities of ICT was archived as raw texts in patent documents [6,7]. Analyzing patent documents effectively is important to acquire technological knowledge, link potential solutions to problems and inspire innovation in the industry [8]. Therefore, exploiting information underlying patent documents has gained increasing interests by researchers, patent analysts, and practitioners [9].

In patent documents of ICT in construction, the hints of communication functionalities are hidden in complicated expressions like how construction data was transmitted through virtual or physical models and how it was coordinated among sites, users or stakeholders [5]. Examples of such expressions include “installation information was transferred from a radio frequency identification (RFID) tag to a construction item” in an RFID patent [10], and “the technology conveys geographic data to display devices that users could manipulate” in a geographic information system (GIS) patent [11]. To make this embedded information more accessible, this study seeks to develop a computer-aided system to automatically identify the communication-oriented entities (CEs) and categorize them into pre-defined types. The task is named as entity recognition in natural language processing (NLP) [12].

Although some patent analysis tools (e.g., TRIZ[1]) have been developed to process patent documents, these approaches aim for general purposes and are limited in specific problem solving [13]. Entity recognition offers a way to analyze patents based on customized problems or interests. An entity is a category of phrases that have similar properties, including rigid designators or members of a semantic class [14]. Mostly, the entities are “names” (e.g., drug names, disease names, chemical names) [14]. They usually have highly distinguishable spellings (e.g., chemistry entity “Deuterium” can be easily recognized due to its unique combination of characters and the capitalized initial letter [15]). However, recognizing CEs from the patent documents of ICT in

[1] TRIZ is the acronym for the “Theory of Inventive Problem Solving” in Russian, which is a tool for patent analysis. See details in https://www.triz.org/triz.

construction is a more complicated task. There are two main technical challenges. One is the ambiguity of CEs. An entity is ambiguous if its spellings appear as an entity at one position, and appear as a different entity type at another [16]. Communication functions in the patents are expressed by not only mixtures of unique technical terms that appear with distinguishable spellings, but also words that are typically normal terms [17]. Thus, for recognizing ambiguous entities, it is important to incorporate the contextual information surrounding the candidate entities to discern their relevancy. Another challenge is the unknown of entities (entities that appear in testing set but not in training set). The previous studies attempted to address these problems by using additional linguistic materials, such as lexicons, dictionaries, gazetteers, ontologies, knowledge graphs [18-21]. However, due to the unavoidable limitations in the coverage of lexical databases, these problems remain critical [22].

This study resorts to deep learning techniques to utilize the contextual information for recognizing the ambiguous and unknown CEs from the patents of ICT in construction. Rather than focusing on word-level information, a deep learning method can enhance the understanding of entities by incorporating surrounding texts. As recognized deep learning approaches, the recurrent neural networks (RNN)-based models, such as long short-term memory (LSTM) and gated recurrent unit (GRU), have been widely adopted in many NLP tasks, including entity recognition, text classification, sentiment analysis, and machine translation [23,24]. In these models, bi-directional structures and convolutional neural networks (CNN) were adopted to achieve improved performance [25]. However, despite the elaborate architectures, the RNN-based models have limitations in addressing long-term dependencies. A deep learning model of the Transformer-based neural networks (TBNN) was adopted instead in this study to remedy this deficiency. Proposed in

2017 by Google AI team [26], the Transformer can enable the so-called "self-attention" mechanism that computes the contextual representations in parallel rather than in sequence [27], enabling a more effective approach to memorize both long and short term dependencies compared with the RNN-based models. Previous methods used for recognizing communication functionalities from ICT patents were mostly manual searching, which are labor-intensive and time-consuming [28,29]. The TBNN model developed in this study provides an efficient alternative. Also, It has its merits in utilizing contextual information, which is an important advancement for computer-aided systems to achieve intelligence in NLP tasks [16].

The research procedure is shown in Fig. 1. First, based on the literature review, the main technical challenges were identified and the classes of CEs for recognition were illustrated. Second, the architecture of the proposed TBNN was illustrated in detail. Third, the validation of the model was conducted using the training and testing instances. Finally, the results and findings were discussed to report the performance of the proposed model compared with the baseline model.

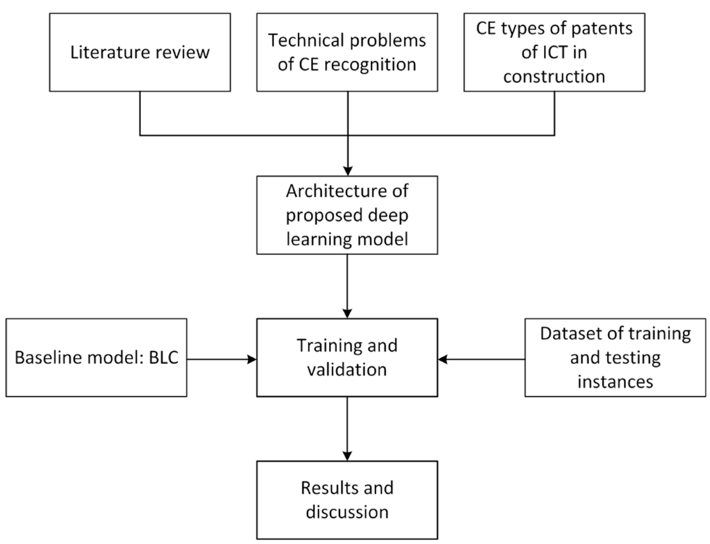

Fig. 1. Workflow of the research.

2. Review of relevant research

2.1 Overview of entity recognition

Entity is an NLP concept that was first introduced in 1996 [18]. An entity is a phrase representing the elements that have similar properties. Entities are rigid designators or members of a semantic class that can be characterized by specific purposes [14]. Generally, entity recognition is used to automatically identify names of people, locations, and organizations using information extraction techniques. At the beginning, such a task was called "Named Entity Recognition". It was rapidly adopted in different fields. For instance, in the dietary research, recognizing entities of food and nutrient gained increasing interests [20]; in the chemical and life science, gene and protein are important entities [30]. Along with the proliferated applications, the ambiguity of entities was soon recognized as a central issue, which can substantially decrease the accuracy of entity recognition results [16].

2.2 Entity recognition models

Over the last two decades, a large and growing number of models have been developed for entity recognition, which can be mainly categorized into two groups: rule-based [20,31-35] and learning-based models [14,36-39].

2.2.1 Rule-based models

The rule-based models usually rely on man-made rules, including lexical attributes and vocabularies. Lexical attributes concern word-level properties [12]. Digit pattern is one typical lexical attribute. It contains information such as data, intervals, and statistics. For example, four digits normally stand for an expression of a year. Similarly, one or two digits usually represent a date [31]. Morphological attribute is another type of lexical attributes. For example, language entities are often ended with "ish", such as Spanish and Danish [32]. Using rules based on lexical attributes can achieve acceptable accuracy. However, the establishment of these rules requires expertise and tremendous efforts, which is expensive to accomplish [12].

Using rules based on vocabularies are also called terminology-driven or dictionary-based methods [20]. These methods recognize entities through matching relevant text with a pre-defined thesaurus consisting of a range of terminologies and their relations [33]. Such approaches sometimes lead to poor performance because of the inevitable ignorance of synonyms [34]. The main problem of the vocabulary-based rules comes from the incomprehensiveness of pre-defined corpora, which can cause dissatisfactory results due to the omission of entities [20].

2.2.2 Learning-based models

The learning-based models employ machine learning algorithms to automatically recognize entities using the patterns learned from training instances [12]. Regarding entity recognition, there are two types of learning-based models: supervised and semi-supervised [14]. Over the last two decades, a number of machine learning algorithms have been used in entity recognition, such as Support Vector Machine [36], Conditional Random Field (CRF) [37], Hidden Markov Model [38], and Maximum Entropy Markov Model [39]. The main drawback associated with these algorithms was the requirement of a large amount of annotated data, which increased human intervention in feature selection [37-39].

In response to this problem, deep learning techniques have been employed in entity recognition. It shows prominent performance in many NLP tasks and does not need manual feature selection as machine learning. Deep learning models are generally organized as multi-layer neural networks, each of which consists of neurons, receiving signals from the former layer and passing converted signals by activation functions to the subsequent layer [40]. These layers of neural networks, as a whole, can address highly non-linear associations between representations and outputs [41]. Most of the deep learning models used for entity recognition are developed based on RNN [42,43]. Fig. 2 displays the architecture of an RNN-based model for entity recognition. The model generally follows a structure framed by “word embedding”, “main recurrent neural networks” and “CRF”.

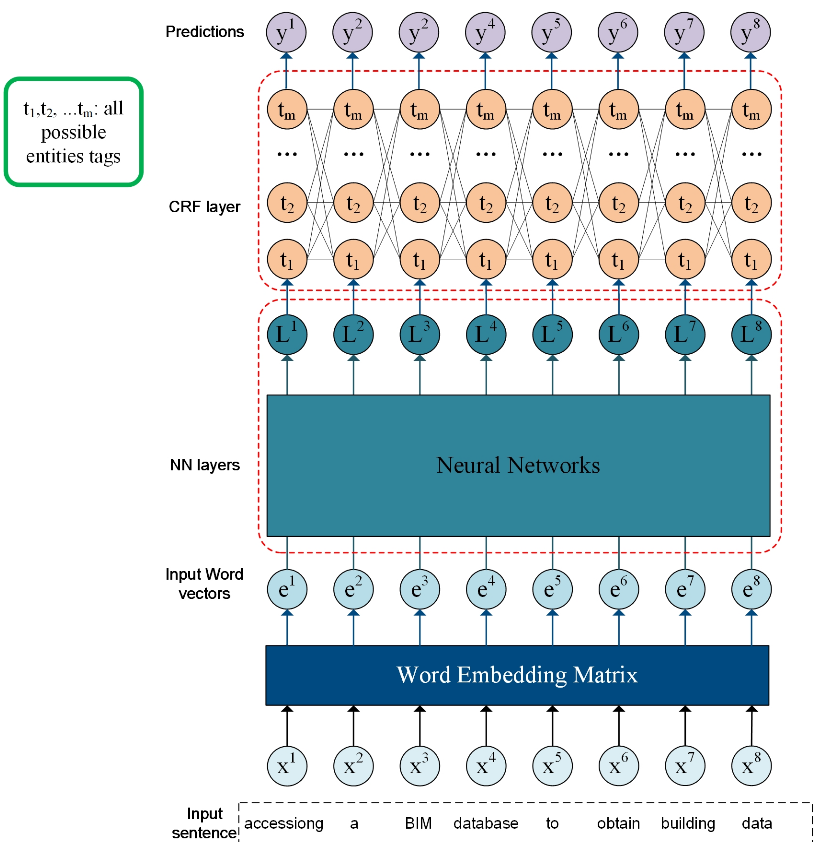

Fig. 2. Typical structure of RNN-based models for entity recognition.

However, the performance of the RNN-based models remains unsatisfactory due to its basic nature of sequential computation. RNN generates a sequence of hidden state values L^t according to previous hidden value L^{t-1} and input value at position t (e^t) [25]. This sequential computation style prevents parallelization in the training process, and thus prevents further utilization of advanced hardware. Without parallelization, the computing for long sequences would take a considerable amount of time due to the limited use of batching across examples [26].

2.3 Entity recognition in CEM studies

In the Construction Engineering and Management (CEM) domain, an increasing number of research starts to apply entity recognition to process textual data in addressing various management issues (i.e., [44-47]). Mostly, traditional rule-based models were adopted using pre-defined digital dictionaries established by experts [48-50]. Such methods are usually labor-intensive and time-consuming, as well as suffer limited coverage of pre-defined corpora [22].

Several efforts have been made in the CEM domain to improve the models for entity recognition [22,51]. For example, Zhang and El-Gohary [51] developed an automatic approach to extract Building Information Modeling (BIM) entities from documents. That study integrated manual rules and a pre-defined lexical database for entity recognition. Specifically, the rules that were established based on part-of-speech patterns and an external word vocabulary were used to extract entities, and the lexical database of WordNet was employed to classify the entities. Similarly, Le and Jeong [22] developed the vocabularies as rules to recognize transportation entities. Such rule-based methods can only identify pre-defined entities, but neglect unknown ones. Also, they were mostly reported with a poor performance in discerning ambiguous entities [16].

Based on the review of relevant research, the study employed deep learning instead of the traditional machine learning algorithms for CE recognition. There are two main reasons: (1) deep learning can draw representations from unstructured text data based on the architecture of neural networks without pre-engineered features that are necessary for traditional algorithms [52,53]; and (2) it can address highly non-linear associations between representations and outputs through the neurons and activation functions in each layer of the neural networks [41].

3. Definition of CE classes

CEs refer to the information units that describe communication functionalities in the patents of ICT in construction. They present approaches of virtual or physical transmission of data, or of data coordination among sites, users or stakeholders [5]. For example, the sentence "sensing the material information through RFID tags" indicates that the RFID technology can be applied to timely transmit information on construction materials [4]. This communication functionality involves two important entities: "material information" and "RFID tags". The former is the information to be transferred, and the latter is the device to send and receive the information.

Based on the specific patterns of ICT in construction as well as the review of relevant literature [5], this study defines three CEs in describing communication functionalities. They are transferred information (TI), communication models (CM) and communication subjects (CS). (See Table 1 for detailed descriptions and examples). Among them, TI refers to the type of information for transmission, for example, the geographic locations. CM refers to the software or equipment used to transmit the information, which can be either virtual or physical. For example, a BIM database is a virtual platform to store, receive and send building data, while an RFID tag is a physical device to store and send information of building components. At last, CS is the people or organizations that involved in the communication process.

Table 1 Description and examples of CE classes for ICT in construction.

CE classes	Description & Examples
TI	Information the ICT mainly conveys, transmits, manipulates, or receive and always in a digital form • The apparatus for editing the 3D building data includes an input unit configured to obtain **3D scan data of a building** • The building's developer sends a **design order** for the building to a construction design office • An estimation engine processes the **aerial image** at a plurality of angles to automatically identify a plurality
CM	Software or equipment that is used to convey the transferred information. CM could be virtual or physical, which could be accessed and manipulated remotely. • The construction operation system comprises a **photodetection sensor** for receiving light beams from the rotary laser irradiating systems • A first hierarchical data structure generated by the mobile device is received at the **first machine** • Exemplary systems and methods include **marking devices** that generate, store and/or transmit electronic records of marking information
CS	People or organizations that participate in communication activities. CS is always the people the transferred information would be delivered to in the context of construction. • The **developer** of the building sends a design order for the building to a construction design office • The design environment supports multi-modal input, side-by-side layout of the stored documents, access permissions for **users** of the design environment • The method comprising: (a) receiving, into the computing device, an input from a **user** (either a **person** or an **automated program interface**)

4. The proposed deep learning model

4.1 The objectives of the proposed model

The developed TBNN model is to utilize contextual information to automatically identify and classify CEs out of patent documents, addressing the aforementioned problems in recognizing ambiguity and unknown of entities. As it is shown in Figure 3, two examples of CEs extractions show the utilization of contextual information in recognizing ambiguous entities. In Fig. 3 (a), the entity "building data" was recognized as TI, because the surrounding text indicated that the "building data" is a type of information that can be transferred remotely. In another case in Fig.3(b), the "building data" was recognized as a normal phrase (labeled as "O") because the model found it is not used for communication based on the surrounding text.

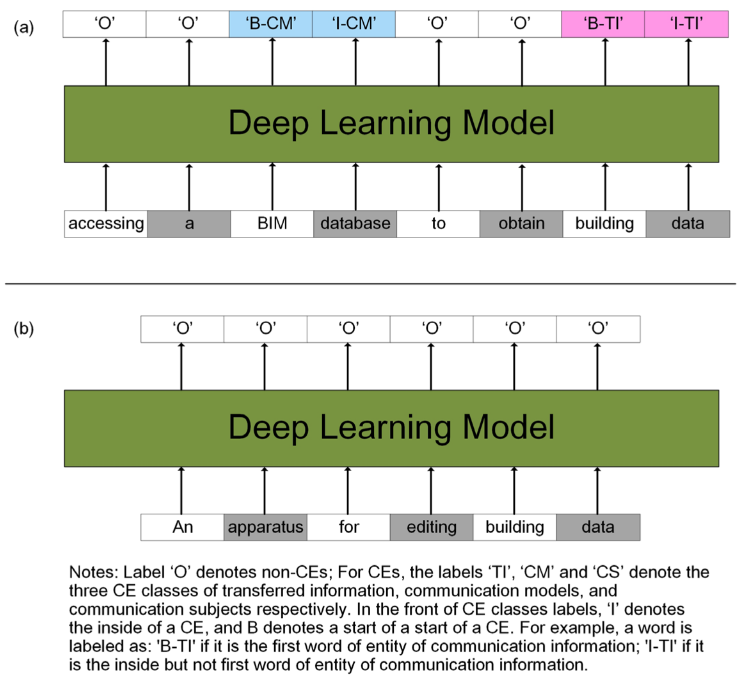

Fig. 3. The inputs and outputs of the desired model for CE recognition.

4.2 The structure of the model

The overall structure of the TBNN was presented in Fig. 4. Instead of using a sequential structure as RNN-based models, it has a parallel system [27,54]. The major components of TBNN include Wordpiece tokenization, token and position embedding, and multi-head self-attention.

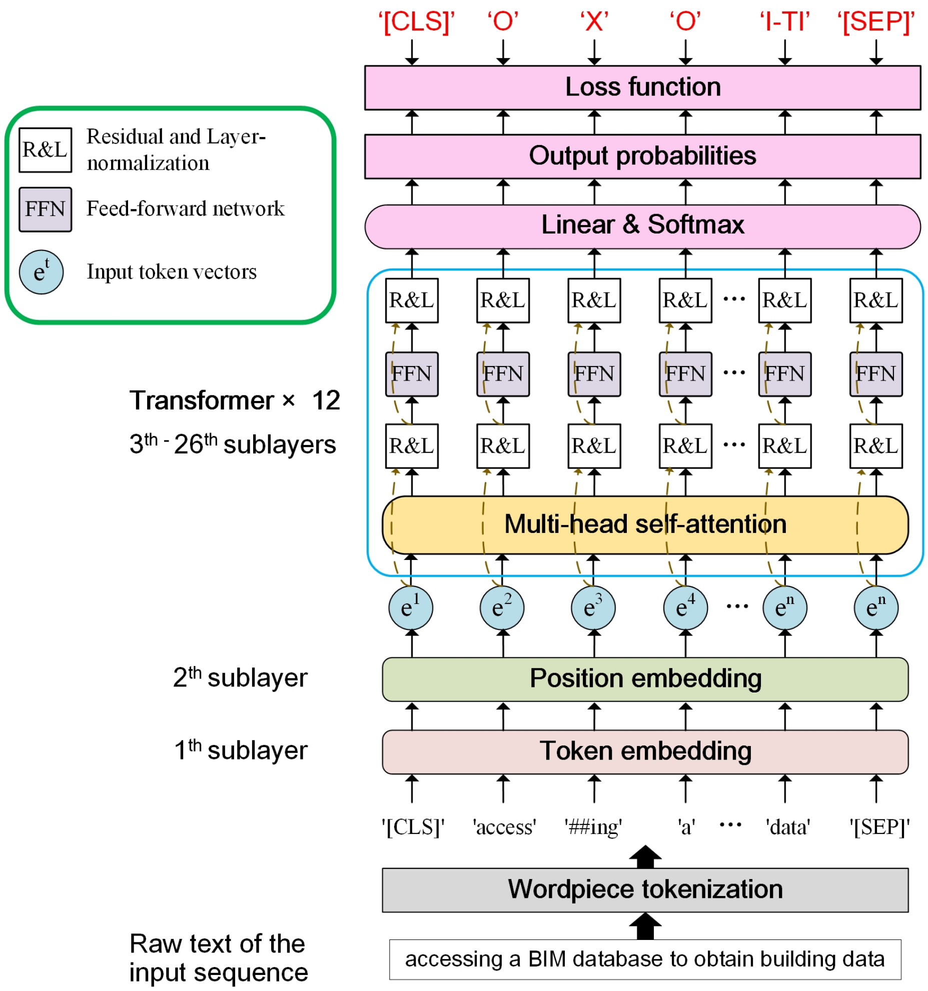

Fig. 4. The overall neural network structure of TBNN.

4.2.1 Wordpiece tokenization

Before feeding into the model, Wordpiece tokenization is used to split the words of input sequences into sub-word units ("word-pieces") that can be small as a letter or large as a complete word [55]. Algorithm 1 outlines the core idea of Wordpiece tokenization, which selects minimal segmented word-pieces that can make combinations of the words [56] (as for the details of the Wordpiece, please see Heinzerling and Strube [3]). It can use a relatively small size vocabulary of word-pieces to represent almost infinite words (this study selects a vocabulary of 30,522 word-pieces). Using

Wordpiece tokenization is essential for the model to process unknown entities. Those entities, although do not appear in the training dataset, can be decomposed into word-pieces and fed into the model for prediction.

Algorithm 1: Wordpiece tokenization

```
def statistic(vocabulary):
    pairs_words = coll.defaultdict(int)
    for token, count in vocabulary.items():
        characters = token.split()
        for i in range(len(characters) - 1):
            pairs_words[characters[i], characters[i + 1]] += count
    return pairs_words

def merge_pair(pair, input_vocabulary):
    output_vocabulary = dict()
    Byte_encoding = re.escape(' '.join(pair))

    p = re.compile(r'(?<!\S)' + Byte_encoding + r'(?!\S)')
    for word in input_vocabulary:
        w_out = p.sub(''.join(pair), word)
        output_vocabulary[w_out] = input_vocabulary[word]
    return output_vocabulary
```

Input: a token vocabulary V and the corresponding occurrence times, and the number for merge times.

Begin

```
for i in range(num_merges):
    pairs = statistic(vocabulary)
    best = max(pairs, key=pairs.get)
    vocabulary = merge_pair(best, vocabulary)
    a = a + 1
```

End

4.2.2 Token and position embedding

Token embedding concerns the information of the tokens' identities, which was widely used in NLP studies. Each of the resulting word-pieces would be converted into numerical vectors to represent their identities through a token embedding matrix $D \in \mathbb{R}^{|v| \times |d|}$ (in this study, $|v| = 30{,}522$, $|d| = 512$). Compared with RNN-based models, TBNN has to independently embed the position information due to the parallel structure of the neural networks. This study utilizes a sinusoidal function as the positional embedding model, because it can memorize the position information for a much longer sequence by using relatively fewer parameters [26]. The position embedding algorithm is represented as Eq. (1).

$$\mathrm{PE}_{i,j} = \begin{cases} \sin\left(\frac{i}{10000^{\frac{j}{d_{ed}}}}\right), if\ j\ is\ even \\ \cos\left(\frac{i}{10000^{\frac{j-1}{d_{ed}}}}\right), if\ j\ is\ odd \end{cases} \tag{1}$$

where i denotes the position for the token to be embedded, and j denotes the dimension of the word embedding.

4.2.3 Multi-head self-attention

The 3rd to 26th layers are stacked 12 transformers, and each of them consists of a multi-head self-attention and a point-wise feed-forward neural network. The multi-head self-attention is a linear projection of multiple self-attention neural networks. Self-attention in the Transformer plays an important role in understanding contextual information. Its key capability is to determine how much attention should be paid to useful inputs when determining an output [23,26]. An output of self-attention is called a "contextual representation", reflecting the word's meaning used in the context [26]. Moreover, self-attention enables parallel computation, effectively reducing the computation burden.

To illustrate the self-attention mechanism, Fig. 5 depicts the process to compute the contextual representation for an input sequence “accessing a BIM database to obtain building data”. After tokenization by Wordpiece, the sequence splits into 12 word-pieces. A word would keep its original label for its first word-piece, and the others are labeled as "X". The outputs of self-attention are the context matrix Z, in which each element is computed as follows:

$$z^{(t)} = \sum_{t'=1}^{n} \mathrm{a}_{t,t'}(\mathrm{x}_{t'} W^{v}) \tag{2}$$

$$\mathrm{a}_{t,t'} = \frac{\exp(\mathrm{r}_{t,t'})}{\sum_{t'=1}^{n} \exp(\mathrm{r}_{t,t'})} \tag{3}$$

$$\mathrm{r}_{t,t'} = \frac{e^{(t)} W^{Q} \left(e^{(t')} W^{K}\right)^{T}}{\sqrt{dc}} \tag{4}$$

where:

- t is the target position that is intended to compute an output $z^{(t)}$ corresponding to the input $e^{(t)}$ by using self-attention.

- t' denotes the position from which the attention should be drawn to the target position t

- $W^{Q}, W^{K}, W^{V} \in \mathbb{R}^{\mathrm{dm} \times dc}$. W^{Q}, W^{K} *and* W^{V} are the query, key and value memory matrix respectively, fully connected with the whole deep learning model and the elements in the matrices are parameters to be estimated during the feed-forward and back-propagation processes via stochastic gradient descent.

- $\mathrm{r}_{tt'}$ is an energy score from $e^{(t')}$ to $e^{(t)}$, achieved by a scaled dot product operation. $\mathrm{r}_{tt'}$ reflects how much attention of the input $e^{(t')}$ with respect to $e^{(t)}$.

- $\mathrm{a}_{t,t'}$ refers to the normalized attention score denoting how much attention should be paid to input $e^{(t')}$. All the attention scores form an attention matrix A in which each row consists of coefficients (sum up to 1) representing the normalized attention weights.

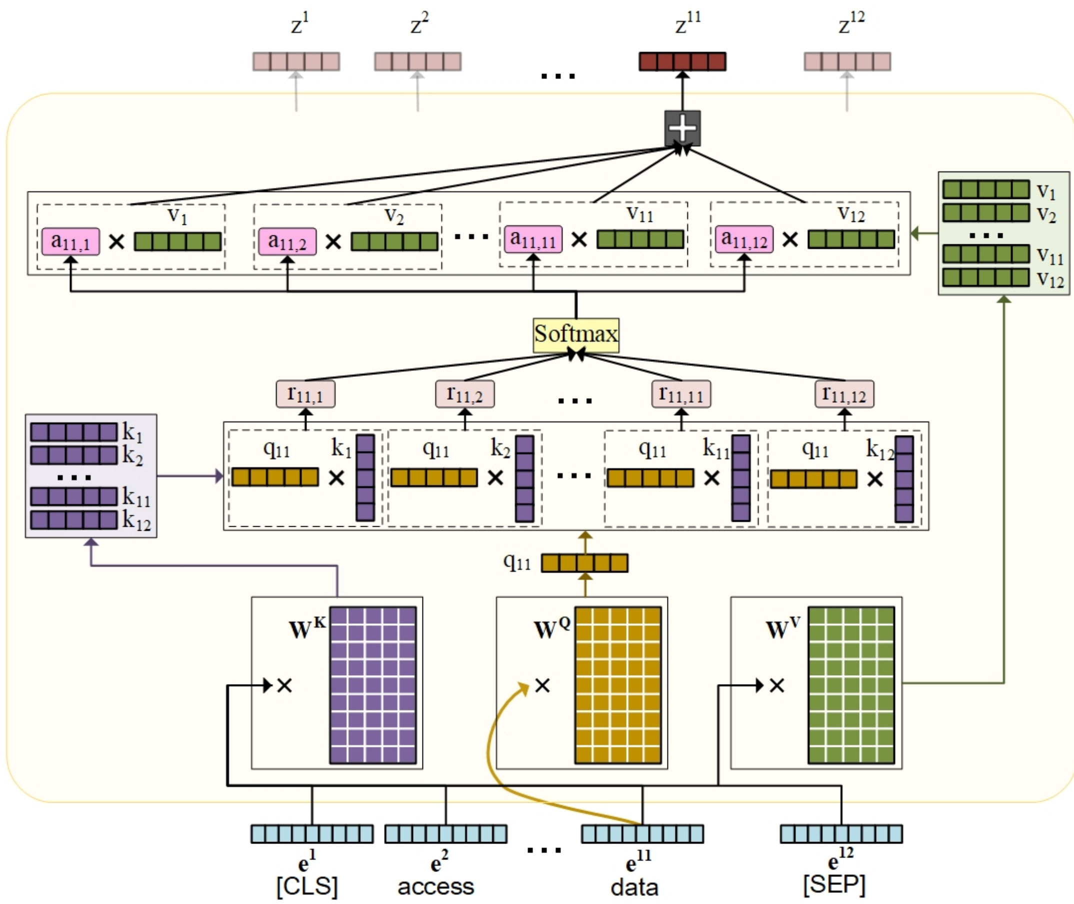

Fig. 5. Computation process of z_{11} by self-attention.

The multi-head self-attention neural network would be fed into a fully connected sublayer of point-wise feed-forward networks (FFN), which treats each position independently and identically. It consists of two linear transformations and an activation function:

$$FFN(x) = \max(0, xW_1 + b_1)\, W_2 + b_1 \tag{5}$$

The outputs of the Transformers would be connected with a linear and a softmax neural network to generate the possibilities for each of the labels. In the training process, the cross-entropy is used as the loss-function to compute the gross gradients for the back-propagation process.

5. Empirical validation

This section reports the validation results of the proposed model compared with the baseline model. This study selected the bi-directional LSTM with CNN (abbreviated as BLC) as the baseline model, which is one of the most typical and outperformed deep learning models for entity recognition [57].

5.1 Description of training and testing datasets

The primary data source for model training and testing is extracted from Wu et al.'s study [58]. The paper developed a binary classifier to automatically screen patents of ICT in construction from United States Patent and Trademark Office (USPTO)[2] (please refer to [58] for details). The screened patents in the study contained not only ICT specifically designed for the construction industry, but also technologies for general communication scenarios. Therefore, the irrelevant patents were eliminated manually. A collection of 392 patents was obtained as the primary dataset. Furthermore, 180 patents out of the primary dataset were randomly selected for annotation. The patents were annotated with titles, abstracts, and first claims using a web-based tool Doccano[3]. The titles and abstracts provide brief and summarized specifications about the technical disclosure. The claims define the patents' protection rights, and the first ones always describe the technical

[2] https://www.uspto.gov

[3] https://github.com/chakki-works/doccano

boundaries [59]. Overall, 2191 CEs were tagged in the 414 sentences. Following previous literature (e.g., [60,61]) that analogously drew upon undersampling for selecting the training instances, this study randomly selected the same number of sentences without any CE tags from the database. The resulting collection comprised 824 sentences. The descriptive statistics of different items in the resulting collection is shown in Table 2.

Table 2
Descriptive statistics of different items in the resulting collection.

ID	Item	Number	Percent of all CEs
1	Total sentences	824	/
2	Annotated sentences	412	/
3	Total words	63765	/
4	Total labels	4392	/
5	Total occurrence of CE	2191	/
6	Total number of CE	1028	/
7	Occurrence of TI	1055	49.32%
8	Occurrence of CM	857	40.07%
9	Occurrence of CS	227	10.61%
10	Number of TI	571	55.54%
11	Number of CM	375	36.48%
12	Number of CS	82	7.98%

This study used k-fold cross-validation to evaluate the performance by setting k as 10. All the instances were randomly divided into 10 folds. For each training round, nine folds consisting of the training and testing collections were made from the rest one, as shown in Fig. 6. In addition, in order to further validate the application of the model in a different text source, this study also retrieved literature relevant to ICT in construction from the Web of Science (WoS). The abstracts were annotated for an additional test collection.

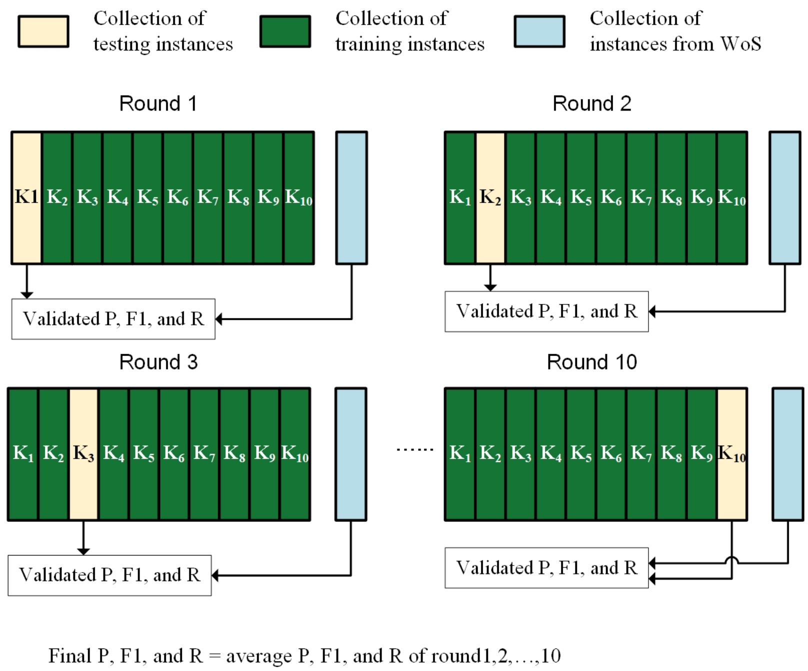

Fig. 6. Illustration for 10-fold validation.

5.2 Pre-trained parameters for Transformers

This study drew upon the pre-training techniques and dumped the pre-trained parameters into the Transformers as initials to make the training process converge quickly. Pre-training techniques were recently developed and experimentally shown to improve the performance of many NLP tasks [62,63].

The pre-training phase in this study has the same structure as the 12 stacked Transformers in TBNN, performing the masked language task that predicts the next words based on the surrounding words [54]. The task follows an unsupervised manner, in which the data is not required to be labeled because the true labels are the input sentence itself in the masked positions. Therefore, a large corpus of data can be used in the pre-training model. Then the trained parameters in the self-attention sublayer of the Transformers can be copied into TBNN. This study used a pre-training model proposed by Google AI team, which contains more than 110M parameters that are learned by Wikipedia corpus[4]. The primary function of the pre-training phase is to provides average contextual representations embedded in the Wikipedia corpus and fine-tunes them through the back-propagation in the training phase of CE recognition, making the TBNN converge rapidly.

5.3 Experiment setup

The experiment setup for training was shown in Table 3. The training programs were implemented based on a workstation with the CPU: Intel(R) Core(TM) i7-7700HQ CPU @2.80Hz 2.81GHz and 16.0G RAM, the GPU: NVIDIA Quadro P4000, 8G. GPU plays a major role in training. To make the TBNN training converge rapidly, this study set the settings based on not only previous studies, but also the nature of written language in the patents of ICT in construction and the computational capacity of the GPU. In specific, because the patents of ICT in construction contain many long sentences, the max sequence length is set as 512 to ensure that all sentences can be fed into the model. The length of the max sequence increases the computation burden for the GPU, and thus the batch size was set as 2 (which is the maximum value after trails) to reduce that burden. In addition, this study set the learning rate and training epochs as 5e-5 and 3 respectively, which were reported as the optimum values when using the pre-trained model.

Table 3
Experiment Setup for TBNN training.

Model settings	
Number of transformers	12
Dimension of WordPiece tokens	512
Number of attention heads	12
Maximum number of hidden states	768
Training settings	
Max sequence length	512
Batch size	2

[4] https://github.com/google-research/bert

Learning rate	5e-5
Training epochs	3

5.4 Validation metrics

This study used precision, recall, F-score [64] as the performance measurements based on true positives (TP), false positives (FP) and false negatives (FN). TP and FP represent, respectively, the numbers of instances that the model correctly and incorrectly predicts. While FN is the number of instances that the model fails to predict. Based on TP, FP and FN, the precision, recall, and F-score were computed by:

$$\mathrm{P} = \frac{TP}{TP+FP}, \mathrm{R} = \frac{TP}{TP+FN}, \mathrm{F1} = \frac{2\times P\times R}{P+R} \tag{6}$$

Since there were three CE classes to be recognized, the number of TP, FP, and FN were counted by three CE classes respectively using the following formulations:

$$\begin{cases} TP_{Total} = TP_{TI} + TP_{CM} + TP_{CS} \\ FN_{Total} = FN_{TI} + FN_{CM} + FN_{CS} \\ FP_{Total} = FP_{TI} + FP_{CM} + FP_{CS} \end{cases} \tag{7}$$

5.5 Validation results

5.5.1 Overall results

This study evaluates the performance of the proposed TBNN against BLC based on the 10-fold instances. BLC is built upon an RNN-based architecture, consisting of a bi-directional LSTM layer, a CNN layer, and a CRF layer. The validation results reveal a superior performance of TBNN compared with BLC (see Table 4). TBNN outperformed BLC in all the training rounds over the two different testing collections. In fact, TBNN yields better performance in all the training rounds than the best round of BLC. TBNN also outperformed BLC over almost all the CE classes (an exception is CS in round 9 of the WoS test).

Table 4
Performance of TBNN against BLC over the 10 training rounds.

						Baseline model: BLC						**Proposed model: TBNN**					
		Number of instances				**All instances**			**CE labels (F1)**			**All instances**			**CE labels (F1)**		
	Round	*TI*	*CM*	*CS*	*All*	F1	P	R	*TI*	*CM*	*CS*	F1	P	R	*TI*	*CM*	*CS*
USPTO	1	131	155	38	324	0.67	0.737	0.614	0.657	0.695	0.600	0.856	0.884	0.83	0.875	0.871	0.727
	2	105	86	21	212	**0.713**	**0.739**	**0.689**	0.687	0.788	0.541	0.854	0.836	0.873	0.843	0.898	0.723
	3	68	52	12	132	0.593	0.532	0.669	0.599	0.597	0.519	0.798	0.793	0.803	0.788	0.814	0.767
	4	125	101	24	250	0.667	0.687	0.648	0.640	0.698	0.667	0.859	0.855	0.864	0.866	0.880	0.729
	5	102	74	22	198	0.661	0.616	0.712	0.647	0.657	0.728	0.834	0.851	0.818	0.864	0.790	0.838
	6	114	75	19	208	0.693	0.675	0.712	0.717	0.656	0.700	0.86	0.874	0.846	0.823	0.906	0.907
	7	131	116	51	298	0.657	0.644	0.672	0.596	0.721	0.659	0.793	0.775	0.812	0.717	0.843	0.891
	8	71	45	18	134	0.644	0.585	0.716	0.639	0.667	0.583	0.85	0.864	0.836	0.820	0.871	0.914
	9	112	64	37	213	0.676	0.645	0.709	0.645	0.673	0.778	**0.882**	**0.864**	**0.901**	0.853	0.878	0.986
	10	96	89	37	222	0.657	0.665	0.649	0.674	0.675	0.539	0.823	0.822	0.824	0.828	0.855	0.740
WoS	1	45	34	6	85	0.400	0.579	0.306	0.365	0.439	0.000	0.706	0.814	0.624	0.748	0.665	0.625
	2					0.389	0.600	0.287	0.366	0.428	0.286	0.702	0.875	0.586	0.665	0.760	0.667
	3					0.421	0.523	0.353	0.366	0.501	0.333	0.689	0.808	0.600	0.703	0.701	0.500
	4					0.393	0.594	0.294	0.369	0.463	0.000	0.732	0.842	0.647	0.762	0.719	0.625
	5					0.423	0.590	0.329	0.433	0.424	0.286	0.702	0.870	0.588	0.721	0.698	0.645
	6					0.455	0.526	0.400	0.444	0.485	0.286	0.739	0.842	0.659	0.704	0.828	0.566
	7					0.434	0.595	0.341	0.417	0.443	0.500	0.704	0.851	0.600	0.702	0.717	0.678
	8					0.455	0.492	0.424	0.438	0.514	0.000	**0.862**	**0.933**	**0.800**	0.867	0.846	0.909
	9					**0.482**	**0.582**	**0.412**	0.455	0.494	0.667	0.669	0.848	0.553	0.686	0.673	0.571
	10					0.341	0.500	0.259	0.288	0.444	0.000	0.766	0.873	0.682	0.754	0.791	0.727

Notes: The highest values for each model over both testing collection are in bold

The heat map in Fig. 7 shows the confusion matrixes of the CE classes over the two models. The numbers are obtained by combining all the testing instances in the ten training rounds, constituting the whole original annotated datasets. The sum of the row indicates the number of true labels of all the CE classes (TP+FN). The diagonal elements are the number of correctly predicted instances of the corresponding CE class. Two major findings were found. Firstly, both models have rarely predicted a CE class incorrectly as another. This result also supports the classifications of CEs. Secondly, compared with BLC, TBNN is less likely to incorrectly predict CEs as normal words (manifested by higher numbers in the last column in Fig. 7(b)), nor incorrectly predict normal words as CEs (manifested by higher numbers in the last row in Fig. 7(b)). This result validates a superior performance of TBNN in discerning CEs in patent documents.

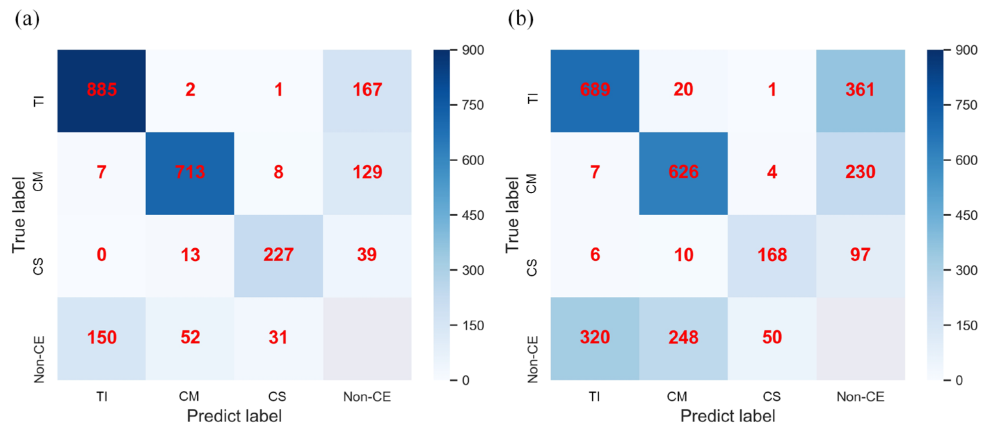

Fig. 7. Heat maps for confusion matrixes: (a) TBNN, (b) BLC. The value v_{ij} corresponds to the number of CE class i that were predicted as CE class j.

Table 5 compares TBNN with BLC in terms of the average validation values of all the training rounds. All the validation indexes of TBNN yielded at least 15% higher than BLC over UPSTO data. Table 5 shows a greater performance of TBNN in testing the WoS literature, indicating that TBNN is more compatible when the training and testing data were from different sources (e.g., training instances from UPSTO and testing instances from WoS).

Table 5
Comparison of BLC and TBNN over the average performance value over the 10 training rounds.

	Baseline model: BLC			Proposed model: TBNN		
	F1	P	R	F1	P	R
USPTO	0.654	0.631	0.683	0.841 **(+18.7%)**	0.842 **(+21.1%)**	0.841 **(+15.8%)**
WoS	0.42	0.558	0.341	0.727 **(+30.7%)**	0.855 **(+29.7%)**	0.634 **(+29.3%)**

In addition, for validating TBNN over real project cases, this study also implemented TBNN to recognize CEs from an industry report named “RFID-Enabled BIM Platform for Prefabrication Housing Production in Hong Kong”. The report describes the applications of RIFD-related techniques to enable communication in a public housing project at Tuen Mun, Hong Kong (See Li, et al. [65] for detailed information). The report (6181 words in total) was input into TBNN for CE prediction. The authors manually examined all the CE prediction results. The precision values are shown in Table 6. It can be observed that TBNN got similar performance in real-project reports with the WoS literature. This test validated that the proposed model performs well in recognizing CEs from documents of real problem scenarios.

Table 6
Precision of CE predictions over the report by TBNN.

	TI	**CM**	**CS**	**All**
TP	13	4	21	38
FP	0	3	4	7
TP+FP	13	7	25	45
Precision	1.000	0.571	0.840	0.844

5.5.2 Validation results in recognizing ambiguous entities

As was mentioned in section 1, a CE can be an ambiguous entity if it appears as a CE at one position and a common noun at another, or appears as different CE types. Table 7 reports the validation results over ambiguous entities. It was found that TBNN performed better in predicting ambiguous entities. The precisions and recall values were higher than BLC by over 13%.

Table 7
Performance TBNN and BLC for ambiguous entities.

Precision		**Recall**		**Number of instances of ambiguous entities**
TBNN	BLC	TBNN	BLC	
0.875	0.753	0.844	0.717	1219

Fig. 8 displays the performance in predicting entities of different ambiguous levels. The horizontal axis describes the CE appearance rate, measured by CE appearance times / total appearance times. The CE appearance times denote the number of times that an ambiguous entity appeared as CEs, and total appearance times represent the total appearance times of the entity. A smaller CE appearance rate indicates a higher ambiguity level. For example, as it is shown in Fig. 9, “image” appears 291 times in the database, only 19 (less than 1%) of them appear as a TI. It has a relatively low CE appearance rate, which means high ambiguity. It leads to lower chances to learn how the surrounding texts determine “image” as a TI. In Fig.8, it shows the change of prediction performance of the two models along with the cumulative percentage of CE appearance rate. The red line represents CE appearance times. It can be found that the smaller CE appearance rate leads to a lower accuracy of both models. Moreover, the gap between the performance of the two models increases as the CE appearance rate decreases. This indicates the superiority of TBNN compared with BLC becomes greater as the ambiguity of entities increased.

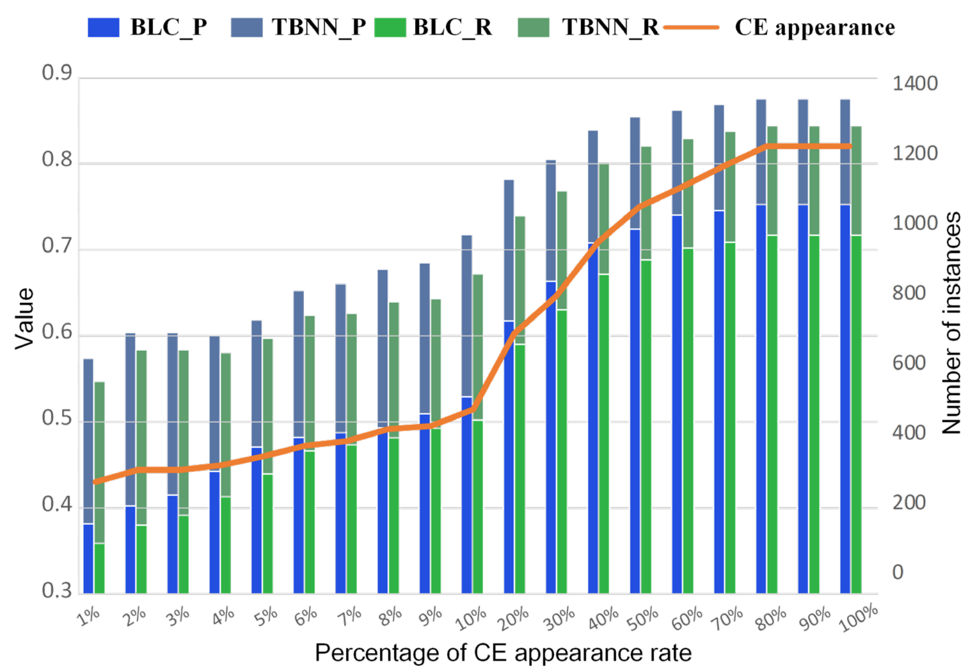

Fig. 8. Performance of TBNN and BLC for ambiguous entities towards different percentages of CE appearance rate.

Fig. 9 plots the performance of the specific ambiguous entities. Three trends can be detected: (1) the ambiguous entities with lower CE appearance rate (i.e., "computer", "sensor", "image", and "display") tend to cause the two models to make incorrect predictions. These entities are much more ambiguous, most of which appear as normal expressions but not CEs in the database. This leads to an extra burden for the two models to discern what contextual information can determine the entity as a CE; (2) the CEs with more specific expressions (i.e., "display device", "facilities map information", "project management system", "user interface", and "location information") tend to experience higher accuracy in both models. More specific expressions convey more word-level and contextual information; (3) TBNN is much better for recognizing ambiguous entities, with higher accuracy in terms of precision and recall.

	Word/phrase	CE class	CE appearance times	Total appearance times	CE appearance rate
1	user	CS	62	264	23.48%
2	computer	CM	24	215	11.16%
3	display device	CM	21	47	44.68%
4	sensor	CM	20	116	17.24%
5	image	TI	19	291	6.53%
6	first participant	CS	16	31	51.61%
7	database	CM	15	77	19.48%
8	computing device	CM	15	66	22.73%
9	position	TI	15	98	15.31%
10	presence	TI	12	27	44.44%
11	selection	TI	11	64	17.19%
12	absence	TI	11	23	47.83%
13	display	CM	11	174	6.32%
14	facilities map information	TI	11	13	84.62%
15	project management system	CM	11	18	61.11%
16	user interface	CM	11	51	21.57%
17	location information	TI	11	16	68.75%
18	second participant	CS	11	25	44.00%
19	input device	CM	10	19	52.63%
20	server	CM	10	68	14.71%

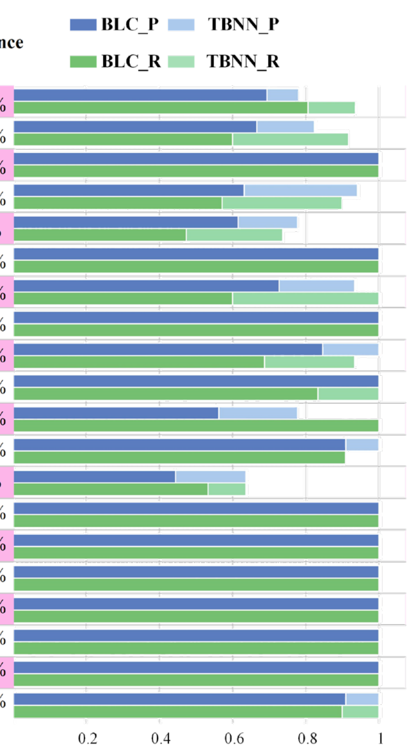

Fig. 9. Performance of TBNN and BLC over the ambiguous entities.

To better illustrate the difference between the recognition process of the two models, Fig. 10 shows three sentences containing the ambiguous entity "user", which may or may not be a CS depending on the contextual information. The ambiguous entity "user" is a CS if the surrounding text indicates that it is involved in a communication context. In case 1, it is not difficult to identify that "user" is a CS, because the former part of this sentence expresses a communication activity involving transferred digital data and communication apparatus. Both models correctly recognized it. Case 2 and case 3 are more complicated, in which TBNN correctly predicted but BLC did not. The word "user" in case 2 is a common word instead of a CS. But when the surrounding context incorporates CEs, BLC predicted it incorrectly as a CS. Case 3 expresses a communication scenario where the user is a participant. The difficulty lies in the vague expression of the communication environment. There are no other CEs in the sentence to provide contextual information. This also misled BLC to an incorrect prediction.

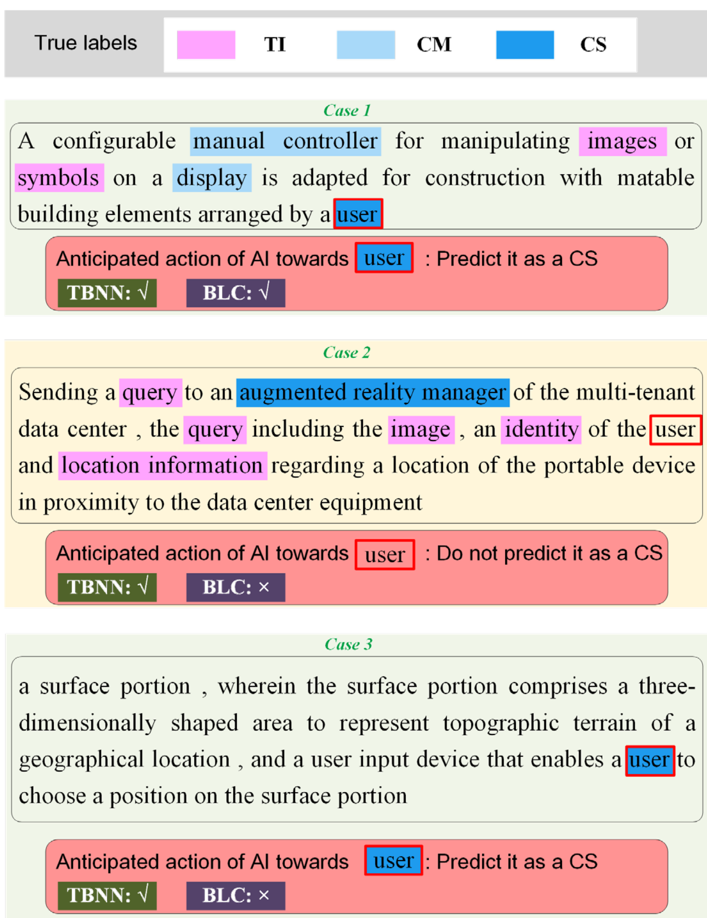

Fig. 10. Examples of recognition of ambiguous entities.

5.5.3 Validation results in recognizing unknown entities

Table 8 reports the validation results over unknown CEs, which were measured by recall values. Other validation measurements, including F-score and precision, are not measurable according to Eq.(6). Because FP is incalculable for unknown entities that appear in testing set but did not in training set. The results show that the performance of both models decreased in predicting unknown CEs. But TBNN's recall value remains as 0.741, which is almost 20% larger than BLC.

Table 8
Recall value of TBNN and BLC for unknown CEs.

TBNN		**BLC**		**Number of instances of unknown CEs**
Total	Unknown CEs	Total	Unknown CEs	
0.841	0.741071	0.683	0.544642857	112

5.5.4 Summary

The tests validated a better capacity of TBNN to utilize the contextual information in recognizing CEs compared with BLC. It can be explained that TBNN has a deeper and thinner neural structure where the dependencies among the input tokens are addressed only by the self-attention mechanism, while the RNN-based structure has only one or two layers of neural networks. In addition, TBNN is found more effective in transmitting gradients, leading to a better learning ability than the RNN-based models. Compared with BLC that sequentially receives and transfers dependencies from a recurrence to the next, TBNN structure draws them parallelly. Furthermore, we also found that TBNN is robust in processing real-life documents other than patents. The paper showed superior performance of TBNN in recognizing CEs from a report of ICT applications in a public housing project.

Compared with similar NLP tasks in previous research, the performance of TBNN proposed in this study is above satisfactory. Especially, using unstructured data as training instances can increase the learning burden for NLP approaches. For example, Baker, et al. [66] used machine learning methods to predict safety outcomes from incident reports and obtained F-score of 0.85. Zhong, et al. [67] developed a deep learning model to classify construction accident narratives and reached F-score of 0.67, whereas Goh and Ubeynarayana [68] employed text mining techniques and got F-score of 0.63 for the same task. As for entity recognition tasks in real-world cases, especially when the raw texts involve a large number of ambiguous entities, the general performance level is relatively low. For example, Zhu and Iglesias [16] developed an approach based on external linguistic materials and achieved F-score range from 0.529 to 0.765 according to different testing datasets. Although some research achieved acceptable precision scores, the proposed model also has its notable performance in dealing with ambiguous and unknown entities.

6. Conclusion

This study proposed a TBNN model to recognize CEs from patents of ICT in construction. It provides an efficient alternative for construction practitioners and stakeholders to better access and comprehend the complex specifications of communication functionalities embedded in the patent documents. The deep learning techniques were employed to overcome the challenges in recognizing ambiguous and unknown entities. The proposed model was based on the Transformer as the basic neural networks to form the self-attention mechanism. It enables the utilization of contextual information. The TBNN structure enables parallel computation for the neurons and the parameters in the same layer, thus being expected with performance improvements compared with traditional RNN-based models. The validation results of multiple empirical tests confirmed this expectation. It can be safely concluded that TBNN has higher performance in CE recognition compared with BLC, especially in the ones with ambiguous and unknown entities.

The model presented in this study offers an effective approach to extract essential information on communication functionalities from the patent documents of ICT in construction. Regardless of diverse writing genres, it can automatically convert an unstructured document into structured and easy-to-perceive units which shows clearly how the ICT can be utilized in construction practices. The recognized CEs, similar to other entity recognition studies, can be used for further NLP applications, such as question answering, text summarization, and information retrieval. Moreover, the model provides an improved approach in applying entity recognition in the field of CEM. As an information extraction approach, entity recognition has not yet been widely adopted to real-world cases like other NLP approaches. Because obtaining satisfactory accuracy entails a large corpus of linguistic materials, especially in the rule-based methods and traditional machine learning models. As for recognizing CEs from patents, such preconditions are too difficult to obtain. The proposed TBNN model, alternatively, utilized contextual information of the input sequences to identify and classify CEs out of common words. The model draws representations from the original input texts based on the architecture of neural networks without any need for pre-engineered features. It also addresses highly non-linear associations between the representations and the outputs (the annotated CE tags) through the nouns and activation functions in each neural network layer.

Two limitations to the presenting study are needed to be acknowledged. First, the deep learning model could automatically identify and classify CEs into pre-defined classes but cannot extract the relations between the recognized CEs. These relations can provide further knowledge on communication functionalities underlying the patent documents. Second, this study employed the pre-training parameters based on Wikipedia materials. Although these pre-trained parameters can draw contextual representations from a widely covered corpus, the specific contexts of ICTs in construction might be overlooked. The model performance could be improved if using materials closely related to ICTs or CEM.

Acknowledgments

This research was supported by the National Natural Science Foundation of China (NSFC) (No. 71771067, No. 71801159 and No. 52078302), the National Natural Science Foundation of Guangdong Province (No. 2018A030310534), and Youth Fund of Humanities and Social Sciences Research of the Ministry of Education (No. 18YJCZH090).

References

[1] P. Mathur, Technological Forms and Ecological Communication: A Theoretical Heuristic, Lexington Books, 2017. ISBN:1498520480.

[2] J.M. Sardroud, Perceptions of automated data collection technology use in the construction industry, Journal of Civil Engineering and Management. 21 (1) (2015), pp. 54-66, https://doi.org/10.3846/13923730.2013.802734.

[3] B. Heinzerling, M. Strube, Bpemb: Tokenization-free pre-trained subword embeddings in 275 languages, Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018), 2018, pp. 2989-2993, https://arxiv.org/pdf/1710.02187 accessed at June 2019.

[4] C.-Y. Cho, S. Kwon, T.-H. Shin, S. Chin, Y.-S. Kim, A development of next generation intelligent construction liftcar toolkit for vertical material movement management, Automation in Construction. 20 (1) (2011), pp. 14-27, https://doi.org/10.1016/j.autcon.2010.07.008.

[5] S. Alsafouri, S.K. Ayer, Review of ICT Implementations for Facilitating Information Flow between Virtual Models and Construction Project Sites, Automation in Construction. 86 (2018), pp. 176-189, https://doi.org/10.1016/j.autcon.2017.10.005.

[6] M. Kim, Y. Park, J. Yoon, Generating patent development maps for technology monitoring

using semantic patent-topic analysis, Computers and Industrial Engineering. 98 (2016), pp. 289-299, https://doi.org/10.1016/j.cie.2016.06.006.

[7] D. Gredel, M. Kramer, B. Bend, Patent-based investment funds as innovation intermediaries for SMEs: In-depth analysis of reciprocal interactions, motives and fallacies, Technovation. 32 (9) (2012), pp. 536-549, https://doi.org/10.1016/j.technovation.2011.09.008.

[8] R.S. Campbell, Patent trends as a technological forecasting tool, World Patent Information. 5 (3) (1983), pp. 137-143, https://doi.org/10.1016/0172-2190(83)90134-5.

[9] C. Camus, R. Brancaleon, Intellectual assets management: from patents to knowledge, World Patent Information. 25 (2) (2003), pp. 155-159, https://doi.org/10.1016/S0172-2190(02)00131-X.

[10] Y. El Ghazali, É. Lefebvre, L.A. Lefebvre, The potential of RFID as an enabler of knowledge management and collaboration for the procurement cycle in the construction industry, Journal of technology management & innovation. 7 (4) (2012), pp. 81-102, http://dx.doi.org/10.4067/S0718-27242012000400007

[11] Y. Deng, J.C.P. Cheng, C. Anumba, Mapping between BIM and 3D GIS in different levels of detail using schema mediation and instance comparison, Automation in Construction. 67 (2016), pp. 1-21, https://doi.org/10.1016/j.autcon.2016.03.006.

[12] D. Nadeau, S. Sekine, A survey of named entity recognition and classification, Lingvisticae Investigationes. 30 (1) (2007), pp. 3-26, https://doi.org/10.1075/li.30.1.03nad.

[13] Z. Li, D. Tate, C. Lane, C. Adams, A framework for automatic TRIZ level of invention estimation of patents using natural language processing, knowledge-transfer and patent citation metrics, Computer-Aided Design. 44 (10) (2012), pp. 987-1010, https://doi.org/10.1016/j.cad.2011.12.006.

[14] A. Goyal, V. Gupta, M. Kumar, Recent Named Entity Recognition and Classification techniques: A systematic review, Computer Science Review. 29 (2018), pp. 21-43, https://doi.org/10.1016/j.cosrev.2018.06.001.

[15] S.A. Akhondi, E. Pons, Z. Afzal, H. van Haagen, B.F.H. Becker, K.M. Hettne, E.M. van Mulligen, J.A. Kors, Chemical entity recognition in patents by combining dictionary-based and statistical approaches, Database. 2016 (2016), https://doi.org/10.1093/database/baw061.

[16] G. Zhu, C.A. Iglesias, Exploiting semantic similarity for named entity disambiguation in knowledge graphs, Expert Systems with Applications. 101 (2018), pp. 8-24, https://doi.org/10.1016/j.eswa.2018.02.011.

[17] W. El‐Ghandour, M. Al‐Hussein, Survey of information technology applications in construction, Construction innovation. 4 (2) (2004), pp. 83-98, https://doi.org/10.1108/14714170410815024.

[18] R. Grishman, B. Sundheim, Message Understanding Conference 6: A Brief History, Conference on Computational Linguistics, 1996, pp. 466-471, https://www.aclweb.org/anthology/C96-1079.pdf accessed at June 2019.

[19] R. Gaizauskas, K. Humphreys, H. Cunningham, Y. Wilks, University of Sheffield: description of the LaSIE system as used for MUC-6, Conference on Message Understanding, 1995, pp. 207-220, https://www.aclweb.org/anthology/M95-1017.pdf accessed at June 2019.

[20] T. Eftimov, B.K. Seljak, P. Korosec, A rule-based named-entity recognition method for

knowledge extraction of evidence based dietary recommendations, PLoS ONE. 12 (6) (2017), https://doi.org/10.1371/journal.pone.0179488.

[21] Y. Ravin, N. Wacholder, Extracting names from natural-language text, Citeseer, 1997.

[22] T.Y. Le, H.D. Jeong, NLP-Based Approach to Semantic Classification of Heterogeneous Transportation Asset Data Terminology, Journal of Computing in Civil Engineering. 31 (6) (2017), https://doi.org/10.1061/(asce)cp.1943-5487.0000701.

[23] D. Bahdanau, K. Cho, Y. Bengio, Neural machine translation by jointly learning to align and translate, arXiv preprint arXiv:1409.0473. (2014).

[24] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, Y. Bengio, Learning phrase representations using RNN encoder-decoder for statistical machine translation, arXiv preprint arXiv:1406.1078. (2014).

[25] X. Ma, E. Hovy, End-to-end Sequence Labeling via Bi-directional LSTM-CNNs-CRF, arXiv preprint arXiv:1603.01354. (2016).

[26] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, Advances in neural information processing systems, 2017, pp. 5998-6008, https://papers.nips.cc/paper/7181-attention-is-all-you-need.pdf accessed at June 2019.

[27] Q. Chen, Z. Zhuo, W. Wang, BERT for Joint Intent Classification and Slot Filling, arXiv preprint arXiv:1902.10909. (2019).

[28] R. Davies, C. Harty, Implementing 'Site BIM': A case study of ICT innovation on a large hospital project, Automation in Construction. 30 (2013), pp. 15-24, https://doi.org/10.1016/j.autcon.2012.11.024.

[29] M. Nourbakhsh, R. Mohamad Zin, J. Irizarry, S. Zolfagharian, M. Gheisari, Mobile application prototype for on-site information management in construction industry, Engineering, Construction and Architectural Management. 19 (5) (2012), pp. 474-494, https://doi.org/10.1108/09699981211259577.

[30] W. Hemati, A. Mehler, LSTMVoter: chemical named entity recognition using a conglomerate of sequence labeling tools, Journal of Cheminformatics. 11 (2019), https://doi.org/10.1186/s13321-018-0327-2.

[31] S. Yu, S. Bai, P. Wu, Description of the Kent Ridge Digital Labs system used for MUC-7, Seventh Message Understanding Conference (MUC-7): Proceedings of a Conference Held in Fairfax, Virginia, April 29-May 1, 1998, 1998, https://www.aclweb.org/anthology/M98-1016.pdf accessed at March 2020.

[32] E. Bick, A Named Entity Recognizer for Danish, Conference on Language Resources and Evaluation, Citeseer, 2004, pp. 305-308, http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.678.9105&rep=rep1&type=pdf accessed at March 2020.

[33] A.R. Aronson, Effective mapping of biomedical text to the UMLS Metathesaurus: the MetaMap program, Proceedings Annual Symposium, Vol. 2001, 2001, pp. 17-21, https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2243666/pdf/procamiasymp00002-0056.pdf accessed at March 2020.

[34] R.A. Miller, F.M. Gieszczykiewicz, J.K. Vries, G.F. Cooper, CHARTLINE: providing bibliographic references relevant to patient charts using the UMLS Metathesaurus Knowledge Sources, Proceedings of Symposium on Computer Applications in Medical Care, 1992, pp. 86-90, https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2248100/pdf/procascamc00003-

0107.pdf accessed at March 2020.

[35] N.S. Kumar, D. Muruganantham, Disambiguating the Twitter Stream Entities and Enhancing the Search Operation Using DBpedia Ontology: Named Entity Disambiguation for Twitter Streams, International Journal of Information Technology and Web Engineering. 11 (2) (2016), pp. 51-62, https://doi.org/10.4018/ijitwe.2016040104.

[36] S.K. Saha, S. Narayan, S. Sarkar, P. Mitra, A composite kernel for named entity recognition, Pattern Recognition Letters. 31 (12) (2010), pp. 1591-1597, https://doi.org/10.1016/j.patrec.2010.05.004.

[37] M. Majumder, U. Barman, R. Prasad, K. Saurabh, S.K. Saha, A Novel Technique for Name Identification from Homeopathy Diagnosis Discussion Forum, Procedia Technology. 6 (2012), pp. 379-386, https://doi.org/10.1016/j.protcy.2012.10.045.

[38] Y. Wang, Z. Yu, L. Chen, Y. Chen, Y. Liu, X. Hu, Y. Jiang, Supervised methods for symptom name recognition in free-text clinical records of traditional Chinese medicine: An empirical study, Journal of Biomedical Informatics. 47 (2014), pp. 91-104, https://doi.org/10.1016/j.jbi.2013.09.008.

[39] S.K. Saha, S. Sarkar, P. Mitra, Feature selection techniques for maximum entropy based biomedical named entity recognition, Journal of Biomedical Informatics. 42 (5) (2009), pp. 905-911, https://doi.org/10.1016/j.jbi.2008.12.012.

[40] M. Riedmiller, Advanced supervised learning in multi-layer perceptrons-from backpropagation to adaptive learning algorithms, Computer standards and interfaces. 16 (3) (1994), pp. 265-278, https://doi.org/10.1016/0920-5489(94)90017-5.

[41] X. Wang, W. Jiang, Z. Luo, Combination of convolutional and recurrent neural network for sentiment analysis of short texts, Proceedings of COLING 2016, the 26th international conference on computational linguistics: Technical papers, 2016, pp. 2428-2437, https://www.aclweb.org/anthology/C16-1229.pdf accessed at March 2020.

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

[43] J.P.C. Chiu, E. Nichols, Named Entity Recognition with Bidirectional LSTM-CNNs, Transactions of the Association for Computational Linguistics. 4 (2016), pp. 357-370, https://doi.org/10.1162/tacl_a_00104.

[44] S. Staub-French, M. Fischer, J. Kunz, B. Paulson, An ontology for relating features with activities to calculate costs, Journal of Computing in Civil Engineering. 17 (4) (2003), pp. 243-254, https://doi.org/10.1061/(ASCE)0887-3801(2003)17:4(243).

[45] K.-Y. Lin, L. Soibelman, Promoting transactions for A/E/C product information, Automation in Construction. 15 (6) (2006), pp. 746-757, https://doi.org/10.1016/j.autcon.2005.09.008.

[46] Y. Rezgui, Ontology-centered knowledge management using information retrieval techniques, Journal of Computing in Civil Engineering. 20 (4) (2006), pp. 261-270, https://doi.org/10.1061/(ASCE)0887-3801(2006)20:4(261).

[47] Z. Zhou, M. Goh Yang, L. Shen, Overview and Analysis of Ontology Studies Supporting Development of the Construction Industry, Journal of Computing in Civil Engineering. 30 (6) (2016), p. 04016026, https://doi.org/10.1061/(ASCE)CP.1943-5487.0000594.

[48] T. El-Diraby, C. Lima, B. Feis, Domain taxonomy for construction concepts: toward a formal ontology for construction knowledge, Journal of Computing in Civil Engineering. 19 (4) (2005), pp. 394-406, https://doi.org/10.1061/(ASCE)0887-3801(2005)19:4(394).

[49] T.E. El-Diraby, K.F. Kashif, Distributed ontology architecture for knowledge management

in highway construction, Journal of Construction Engineering and Management. 131 (5) (2005), pp. 591-603, https://doi.org/10.1061/(asce)0733-9364(2005)131:5(591).

[50] D.P.K. Seedah, C. Choubassi, F. Leite, Ontology for Querying Heterogeneous Data Sources in Freight Transportation, Journal of Computing in Civil Engineering. 30 (4) (2016), https://doi.org/10.1061/(asce)cp.1943-5487.0000548.

[51] J.S. Zhang, N.M. El-Gohary, Extending Building Information Models Semiautomatically Using Semantic Natural Language Processing Techniques, Journal of Computing in Civil Engineering. 30 (5) (2016), https://doi.org/10.1061/(asce)cp.1943-5487.0000536.

[52] N. Mahmoudi, P. Docherty, P. Moscato, Deep neural networks understand investors better, Decision Support Systems. 112 (2018), pp. 23-34, https://doi.org/10.1016/j.dss.2018.06.002.

[53] N. Kalchbrenner, E. Grefenstette, P. Blunsom, A convolutional neural network for modelling sentences, arXiv preprint arXiv:1404.2188. (2014).

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

[55] A. Kannan, Y. Wu, P. Nguyen, T.N. Sainath, Z. Chen, R. Prabhavalkar, An Analysis of Incorporating an External Language Model into a Sequence-to-Sequence Model, 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2018, pp. 1-5828, 10.1109/ICASSP.2018.8462682.

[56] Q.J. Qiu, Z. Xie, L. Wu, W.J. Li, Geoscience keyphrase extraction algorithm using enhanced word embedding, Expert Systems with Applications. 125 (2019), pp. 157-169, https://doi.org/10.1016/j.eswa.2019.02.001.

[57] M. Hofer, A. Kormilitzin, P. Goldberg, A. Nevado-Holgado, Few-shot Learning for Named Entity Recognition in Medical Text, arXiv preprint arXiv:1811.05468. (2018).

[58] H.Q. Wu, G. Shen, X. Lin, M.L. Li, B.Y. Zhang, C.Z.D. Li, Screening patents of ICT in construction using deep learning and NLP techniques, Engineering Construction and Architectural Management. 27 (8) (2020), pp. 1891-1912, https://doi.org/10.1108/ecam-09-2019-0480.

[59] H. Niemann, M.G. Moehrle, J. Frischkorn, Use of a new patent text-mining and visualization method for identifying patenting patterns over time: Concept, method and test application, Technological Forecasting and Social Change. 115 (2017), pp. 210-220, https://doi.org/10.1016/j.techfore.2016.10.004.

[60] A. Garcia-Pablos, M. Cuadros, G. Rigau, W2VLDA: Almost unsupervised system for Aspect Based Sentiment Analysis, Expert Systems with Applications. 91 (2018), pp. 127-137, https://doi.org/10.1016/j.eswa.2017.08.049.

[61] C. Drummond, R.C. Holte, C4. 5, class imbalance, and cost sensitivity: why under-sampling beats over-sampling, Workshop on learning from imbalanced datasets II, Vol. 11, Citeseer, 2003, pp. 1-8, http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.68.6858&rep=rep1&type=pdf accessed at March 2020.

[62] A.M. Dai, Q.V. Le, Semi-supervised sequence learning, Proceedings of the 29th International Conference on Neural Information Processing Systems, 2015, pp. 3079-3087, http://papers.nips.cc/paper/5949-semi-supervised-sequence-learning.pdf accessed at Mach 2020.

[63] M.E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, L. Zettlemoyer, Deep contextualized word representations, arXiv preprint arXiv:1802.05365. (2018).

[64] M. Sokolova, G. Lapalme, A systematic analysis of performance measures for classification tasks, Information Processing & Management. 45 (4) (2009), pp. 427-437, https://doi.org/10.1016/j.ipm.2009.03.002.

[65] X. Li, G.Q. Shen, P. Wu, T. Yue, Integrating Building Information Modeling and Prefabrication Housing Production, Automation in Construction. 100 (2019), pp. 46-60, https://doi.org/10.1016/j.autcon.2018.12.024.

[66] H. Baker, M.R. Hallowell, A.J.P. Tixier, AI-based prediction of independent construction safety outcomes from universal attributes, Automation in Construction. 118 (2020), p. 103146, https://doi.org/10.1016/j.autcon.2020.103146.

[67] B. Zhong, X. Pan, P.E.D. Love, L. Ding, W. Fang, Deep learning and network analysis: Classifying and visualizing accident narratives in construction, Automation in Construction. 113 (2020), p. 103089, https://doi.org/10.1016/j.autcon.2020.103089.

[68] Y.M. Goh, C.U. Ubeynarayana, Construction accident narrative classification: An evaluation of text mining techniques, Accident Analysis & Prevention. 108 (2017), pp. 122-130, https://doi.org/10.1016/j.aap.2017.08.026.