The following publication Wu, H., Shen, G., Lin, X., Li, M., Zhang, B. and Li, C.Z. (2020), "Screening patents of ICT in construction using deep learning and NLP techniques", Engineering, Construction and Architectural Management, Vol. 27 No. 8, pp. 1891-1912 is published by Emerald and is available at https://doi.org/10.1108/ECAM-09-2019-0480

Screening Patents of ICT in Construction Using Deep Learning and NLP Techniques

Abstract

Purpose - This study proposes an approach to solve the fundamental problem in using query-based methods (i.e. searching engines and patent retrieval tools) to screen patents of information and communication technology in construction (ICTC). The fundamental problem is that ICTC incorporates various techniques and thus cannot be simply represented by man-made queries. To investigate this concern, this study develops a binary classifier by utilizing deep learning and NLP techniques to automatically identify whether a patent is relevant to ICTC, thus accurately screening a corpus of ICTC patents.

Design/methodology/approach - This study employs NLP techniques to convert the textual data of patents into numerical vectors. Then, a supervised deep learning model is developed to learn the relations between the input vectors and outputs.

Findings - The validation results indicate that (1) the proposed approach has better performance in screening ICTC patents than traditional machine learning methods; (2) besides the United States Patent and Trademark Office (USPTO) that provides structured and well written patents, the approach could also accurately screen patents form Derwent Innovations Index (DIX), in which patents are written in different genres.

Practical implications - This study contributes a specific collection for ICTC patents, which is not provided by the patent offices.

Originality/value - The proposed approach contributes an alternative manner in gathering a corpus of patents for domains like ICTC that neither exists as a searchable classification in patent

offices, nor being accurately represented by man-made queries.

Keywords: ICT in construction; NLP; Deep learning; Information management.

1. Introduction

1.1. Research background

Information and communication technology (ICT) has been recognized as a key determinant to improve the level of coordination and collaboration in the architectural, engineering, and construction (AEC) industry (Davies and Harty, 2013; Wu et al., 2017). Yet, compared with other industries, the overall adoption rate of ICT in the AEC industry is low (Ahuja et al., 2009), and only a few number of regular and conventional ICTs such as 2D drawings are widely adopted. Regardless of the widely recognized benefits, most of the advanced and novel ICTs applications such as GPS, 4D modelling, BIM and mobiles are still incidentally employed in the industry (Ahuja et al., 2010; Dehlin and Olofsson, 2008; Frits, 2007; Li et al., 2019). One of the major barriers is that construction practitioners always lack technological knowledge about ICTC (Adriaanse et al., 2010; Sardroud, 2015).

Up to 80% technological information is exclusively provide by patents - recognized as one of the most valuable resources for technical analysis (Chiarello et al., 2018; Hoetker and Agarwal, 2007; Terragno, 1979). The content archived in a patent document normally expresses scientific and technological information for the technology application in terms of main machines and approaches involved, basic functions of the application, process whereby the application implements, and solutions to problems (Intarakumnerd and Charoenporn, 2015). Therefore, a corpus of patents that widely covers the inventions of ICTC is a valuable database, not only providing a dictionary for accessing ICTC, but also identifying problems to be solved by the state of art ICTC inventions and recognizing all possible specific embodiments of ICTC (El-Ghandour and Al-Hussein, 2004).

However, such a corpus of ICTC patents does not exist. Table 1 provides the existing patent classes for the AEC industry in the three major patent offices, including World Intellectual Property Organization (WIPO), European Patent Office (EPO), and United States Patent and Trademark Office (USPTO). Table 1 shows that none of the patent offices provide a searchable classification for ICTC. Two offices, WIPO and USPTO provide a specific category of patents that are relevant to the AEC industry, namely E and D25 respectively. The two classes focus on inventions about building materials and fixed construction rather than information and communication technologies.

Table 1 Existing classification schemes in the three major patent offices

Classification Scheme	Organizations	The specific classification of patents related to the AEC industry
International Patent Classification (IPC)	World Intellectual Property Organization (WIPO)	E: Fixed Constructions E01. Construction of roads, railways, or bridges E02. Hydraulic engineering; foundations; soil-shifting E03. Water supply; sewerage E04. Building E05. Locks; keys; window or door fittings; safes E06. Doors, windows, shutters, or roller blinds, in general; ladders E21. Earth or rock drilling; mining E99. Subject matter not otherwise provided for in this section
Cooperative Patent Classification (CPC)	European Patent Office (EPO)	None
United States Patent Classification (USPC)	USPTO	D25: Building units and construction elements 1. Structure 2. Prefabricated unit 3. Stair, ladder, scaffold, or similar support 4. Trellis or treillage unit 5. Architectural stock material

1.2. The problem of retrieving ICTC patents by using query-based methods

In the absence of a searchable classification for ICTC in the patent offices, query-based methods (including the searching engines and other patent retrieval methods) became a possible way for users to retrieve the patents. These query-based methods aim to retrieve all documents that are relevant to a given patent application according to a query. Hence, the accuracy and coverage of retrieval results highly depends on the query (Zhang et al., 2018), which can be formed by a variety of items, such as keywords, citations, authors, granted year, application date, or combinations of them. The core technique lies in the query-based methods is query reformulation - converting the input query into new and more searchable queries (Alberts et al., 2017; Shalaby and Zadrozny, 2019). The query reformulation, including query reduction techniques (Bouadjenek et al., 2015; Mahdabi et al., 2011), query expanding method (mainly by external dictionary and corpus or ontologies) (Azad and Deepak, 2019; Enesi et al., 2018; Tannebaum and Rauber, 2014), semantic-based methods (Girthana and Swamynathan, 2018), metadata-based methods (citations and classification) (Azad and Deepak, 2019; Giachanou et al., 2015; Mahdabi and Crestani, 2014), and interactive methods (Shalaby and Zadrozny, 2018), enriched the query-based methods and have obtained performance improvement in recent studies.

However, gathering the corpus of ICTC patents may not achieve good performance by using the query-based methods, because it is extremely challenging to accurately represent and widely cover the ICTC patents by man-made queries. The patent retrieval tasks, including *prior-art search*, *patentability search* and *infringement search* (Zhang et al., 2018), aim to return a wide coverage of patent documents that are relevant to a patent application according to a query, helping potential patentees check and analyze relevant information before the patent application is granted.

Therefore, the queries are frequently used to represent a specific patent application rather than a set of patents. ICTC, standing for a set of ICT applications that were invented with major embodiments in the AEC industry (Ahuja et al., 2009; Alsafouri and Ayer, 2018), incorporates a number of technologies which may vary with each other (for example, both BIM and RFID are important ICT applications in the AEC industry, but they are totally different technologies). Therefore, completely representing all the ICTC patents by a man-made query leaves a tough task to return accurate results. Moreover, using a query combined by a number of items to represent all the ICTC patents increases the irrelevant instances returned due to polysemy (the same spellings may have two or more different meanings).

This study performs two trails for retrieving ICTC patents from the USPTO website (USPTO, 2007) based on a query combined by a number of items. Table 2 shows the results using this query-based method, and 50 patents were randomly selected to manually check the accuracy - the proportion the ICTC patents occupy all the retrieved patents. Even though a complicated combination of items were used to search ICTC patents, the accuracy is low. Moreover, most of the latent users in the construction practice are non-experts, who may not be able to perform such a searching task that is complex and time-consuming (Liu et al., 2011).

Table 2 Searching results by using the search engine in USPTO

	Querying strategies	**Matched results**	**Accuracy**
Strategy 1	**Query items:** CPC Classification Class and topic (matching input keywords within patent titles, abstracts and descriptions) CPC Classification Class: ICT-related classes, including **H04** - *electric communication technique;* **G06** - *computing, calculating or counting;* **H01P** - *waveguides, resonators, lines, or other devices of the waveguide type;* **H01Q** - *antennas, i.e. radio aerials;* **G01S** - *radio direction-finding, radio navigation,*	Collection 1: 5311 patents	8%

	determining distance or velocity by use of radio waves, locating or presence-detecting by use of the reflection or reradiation of radio waves or analogous arrangements using other waves; **G08B** - *signalling or calling systems, order telegraphs or alarm systems;* **G08C** - *transmission systems for measured values, control or similar signals;* **G11B** - *information storage based on relative movement between record carrier and transducer.* Keywords: AEC domain terms, including *construction project, project management, infrastructure project, civil engineering* and *transportation project.* (Flyvbjerg, 2014; Greiman, 2013; Levitt, 2007; Mok et al., 2015; Zidane et al., 2013)		
Strategy 2	**Query item:** Topic Keywords: ICT-related terms (*Radio frequency identification (RFID), 3D laser scanning, Quick response, NFC, Augmented reality (AR), Mobile computing, Wireless connection (Wi-Fi) and Robotics(drones)*) (Ahuja et al., 2009; Alsafouri and Ayer, 2018; Li et al., 2016) and AEC domain terms	Collection 2: 922 patents	12%

1.3. Research Objectives

Given the aforementioned constraints of existing query-based methods, this study develops a binary classifier to automatically identify whether a patent is relevant to ICTC, and thus accurately screening a corpus of ICTC patents from the primarily searched results containing a number of irrelevant patents. Therefore, this study treats the task of screening ICTC patents as a classification task rather than a retrieval task. A large number of studies have investigated patent classification, and most of them emphasized the use of traditional machine learning (i.e. SVM and Bayes) and text mining techniques (i.e. n-gram and stop-words removal) (Li et al., 2012; Wu et al., 2010). Alternatively, this study resorts to the techniques from the realm of NLP and deep learning. On one hand, NLP techniques provide a smart way to process textual data (Kurdi, 2017), saving time and avoiding personal bias in analysis processes (Agrawal and Henderson, 2002; Bell et al., 2009; Cassetta et al., 2017; Choi et al., 2012; Gwak and Sohn, 2018), especially when the volume of a text is large (Shekarpour et al., 2015; Silva et al., 2016). On the other hand, a deep learning model, Multi-Layer Perceptron (MLP) is developed to learn the relations between the input features and outputs. Deep learning is the most state-of-the-art approach with significant performance improvement in NLP tasks. Compared with the algorithms and statistics of the machine learning models, the deep learning models are organized by multiple layers of neural networks. Each layer consists of neurons, receiving signals from the former layer and passing converted signals by activation functions to the subsequent layer (Riedmiller, 1994). With the multiple layers of neural networks, the whole deep learning model can address highly non-linear associations between the representations and the outputs (Wang et al., 2016), whereas the machine learning algorithms can only examine linear relations.

2. Related work

Several attempts have been made to establish classifiers for automatic patent classification (Chakrabarti et al., 1998; Smith, 2002; Venugopalan and Rai, 2015). Most of the studies, at the beginning, extracted the features from the structured data or metadata, such as keywords and citations (Michel and Bettels, 2001; Perez-Molina, 2018). In the recent decade, scholars prefer using unstructured data (Cambria and White, 2014; Collobert et al., 2011; Gimpel et al., 2011). These studies typically have three key steps: processing the textual data, vectorizing the patents and using machine learning methods to train the models. Focusing on the three key steps, this section describes a synopsis of the relevant literature.

2.1. NLP techniques for processing textual data

A large volume of unformatted texts exist in the current web 2.0 era (Ittoo et al., 2016). The chunk information is mainly unstructured and thus cannot be processed by machine-tractable ways (Cambria and White, 2014) that structured data can be. Processing the unstructured data is regarded as one of the most time-consuming step of text classification tasks (Munková et al., 2013). The major object of the processing is to clean and format the raw textual data, which can largely eliminate noisy features for further vectorization (Haddi et al., 2013). Many NLP tools have been introduced, such as stop and common words removal, tokenization, lemmatization and stemming (Aggarwal and Reddy, 2013). Two typical tools are part-of-speech (POS) and Named Entity Recognition (NER), which can recognize and process syntax information (grammatical meanings) (Collobert et al., 2011; Gimpel et al., 2011) and named entities respectively (Nadeau and Sekine, 2007).

2.2. Vectorizing methods

With regard to the vectorization, a number of algorithms have been developed to convert the textual data into vectors. Bag-of-words (BOW), topic models and subject–action–object (SAO) have been used in recent patent classification studies (Li et al., 2018; Venugopalan and Rai, 2015). Traditional BOW models typically construct the feature space vectors in which each position is occupied by a term or a phrase (Forman, 2002). Its measurements include n-grams, bi-grams, and word frequency to identify phrases from the texts (Onan et al., 2016), depending on how the phrases were counted. Although BOW models are simple and may generate a large number of features, they remain the most effective feature selection method (Mirończuk and Protasiewicz, 2018; Onan et al., 2016). Topic model and subject–action–object (SAO) were mainly developed to solve the high dimension problem, replacing the BOW features by latent topics (Kaplan and Vakili, 2013) or SAO structures (Gerken, 2012).

2.3. Supervised learning models for patent classification

To date, most of the patent classifiers are trained by machine learning models, such as SVM, Naive Bayes, and k-nearest-neighbor (KNN). The accuracy was relatively low in earlier studies (around 70%) (Saiki et al., 2006), but an increasing trend has been observed when feature selection models and NLP techniques were used to extract useful features from unstructured texts, achieving accuracy around 85% (Venugopalan and Rai, 2015; Wu et al., 2010). In the recent decade, the widespread use of deep learning models has led to notable success. Deep learning models have been developed and adopted in a variety of fields, such as natural language understanding, video and image recognition and game of GO (Al Rahhal et al., 2018; Cocarascu and Toni, 2018; Silver et al., 2016). Those deep learning models were always developed with complex and elaborate architecture in which multiple layers of neural networks were well structured. However, only the artificial neural network (ANN) with one layer neurons was applied to patent classification, and the accuracy was relatively low, with 75% (Li et al., 2012).

In addition, most of the research have sought to classify patents into pre-defined classes that already existed in the classification schemes in patent offices, such as IPC and European classification code hierarchy (Fall et al., 2003a; Fall et al., 2003b). Such expositions provide automatic and efficient methods for inventors and examiners to label the new patents with existing classes, but do not provide opportunities to advance the understanding of the real world in the target field. Although using deep learning models may lead to better performance than traditional machine learning models, rare studies employ deep learning models in automatic patent classification tasks.

3. The proposed approach integrating deep learning and NLP techniques

To screen ICTC patents from a patent collection, a binary classifier is developed to classify pieces of patents into two classes: ICTC-related or not. Figure 1 shows the overall procedure of the approach to achieve the classifier. The first step is to collect a database for training, incorporating the full texts of the instances annotated with target labels (the two classes). Then, NLP tools are used to process textual data to achieve clean texts. Based on the processed texts, N-gram and Tf-Idf algorithms are employed for the vectorization to represent each of the patents as a numerical vector that could be fed into the MLP that would be trained by gradient descent in which the hyperparameters are tuned. At last, a validation experiment is conducted by means of k fold cross-validation in two datasets. The succeeding sections discuss these steps.

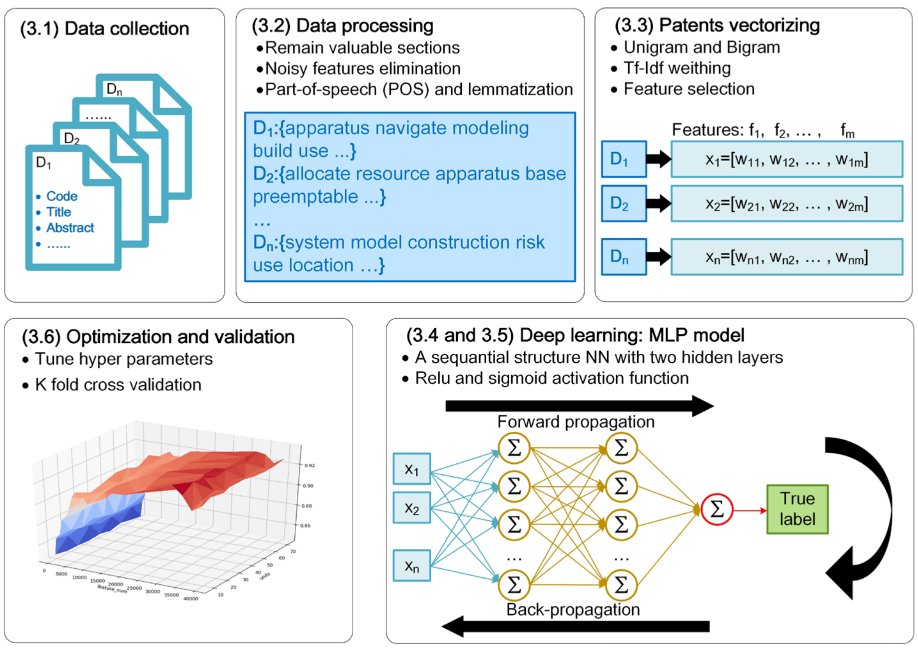


Figure 1 Overall procedure of the approach

3.1. Data collection and annotation

The target of this step is to obtain training data - the full texts of the patents that are manually labeled as *ICTC* or *non-ICTC*. All the required patent text were crawled from USPTO, because (1) USPTO is the largest international patent grant office, and (2) USPTO is recognized as the most representative database to analyze the technological knowledge, providing patents that are well written and structured according to its requirement (Wang, 2018). The authors retrieved the patents on July 30, 2018. Totally, we have collected and annotated 348 patents as the dataset for further training and testing. The detailed processes are described in the following two paragraphs.

Figure 2 depicts the data collection and annotation process, whereby patents were collected and annotated as *ICTC* or *non-ICTC*. As for the *ICTC class*, patents were gathered in the following steps: (1) By querying search strategy 1 in Table 1 (ICT classes and AEC domain terms), 5311 patents were obtained in collection 1. (2) Totally 1500 patents were randomly selected from the 5311 patents. (3) Through the process of manually checking[1], 174 patents were obtained as *ICTC class* from the 1500 patents.

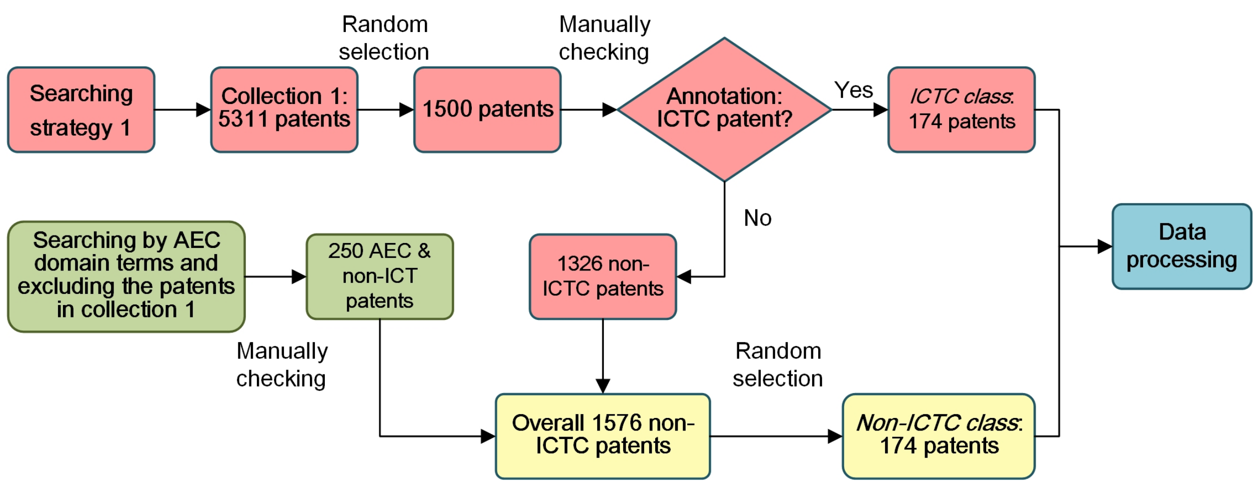


Figure 2 The process of data collection

As for the *non-ICTC class*, the patents were collected from two different sources. One was through the annotation process mentioned above, in which 1326 patents were identified as *non-ICTC* class. The other was obtained from the patents retrieved by searching AEC domain terms and excluding patents in collection 1. This results in a combined collection of 1576 non-ICTC patents and 174 of

[1] The process of manually checking labels a patent as either *ICTC* or *non-ICTC*, performed by three Ph.D. students (their research directions are related to the AEC area) through in-depth reviewing of the title, abstract, claim and description. A patent can be labeled as *ICTC* class if the content expresses that the essence of the technology application is under the ICT scope and the AEC industry is a major embodiment in which the technology application can be implemented. To prevent mistakes as much as possible, two students annotated the patents independently, and the third student would make a judgment when the labels of a patent are inconsistent.

them were randomly selected as training instances for the *non-ICTC class*. The complex collection process has two advantages: (1) The *non-ICTC* class contains not only common ICTs that exclude ICTC patents, but also the technologies of the AEC industry that exclude ICTC patents. This can prevent the data over-fitting, thus generating a more generalized model that is able to distinguish ICTC patents from ICT, as well as from AEC patents; (2) This study uses the negative sampling to make the two classes have the same size, because the balanced size for each class is proven as a key factor to achieve high accuracy in training (Brown and Mues, 2012; Zhao et al., 2015).

3.2. Data processing by NLP techniques

The raw text of each patent contains several sections (i.e., code, title, abstract, CPC classes, inventors and countries and description). Among them, *title*, *abstract* and *claim* are frequently utilized and remained for further analysis in this study because they were recognized as useful items providing basic technological information (Niemann et al., 2017; Venugopalan and Rai, 2015). *Title* and *abstract* convey the essence about the technology, which were always written in a restricted pattern within short content (Lee et al., 2013). In addition, *claim* defines the protection right of the invention, always providing articulated expressions about the technical boundaries and specifications (Niemann et al., 2017).

The selected text of patents is raw data, which is pre-processed by NLP techniques for further analysis. Without pre-processing, the texts would contain a lot of noisy features (in a typical case, the number of features can be close to the number of words in the dictionary of the training instances), thus creating higher-dimension vectors. To process the selected raw text, this study employs three NLP techniques (Figure 3 plots the pre-processing procedure using these techniques): (1) Tokenization. For each raw sentence in the texts, tokenization is utilized to split the sentence into words. In addition, all the words are converted to lowercase and punctuations are removed. Through this step, all the raw sentences would be replaced with sequenced and lowercase words. (2) POS tagging. In this step, each word is tagged with POS tag indicating its syntactic role (i.e. noun, adverb) according to the surrounding words. POS tagging plays a central role in text processing, which could increase the accuracy for lemmatization and stemming (Habash et al., 2009). (3) Lemmatization and stop-words removal. The purpose of this step is to correctly match the words with different forms, such as plural forms for nouns and presenting and past forms for verbs. Lemmatization transforms the different forms into the stem forms (root words). However, lemmatization may generate a number of mistakes without POS tagging. For example, "modeling" may be a present participle of a verb (with lemma "model") or a noun (with lemma "modeling") according to the context, and the lemma of noun "modeling" would be wrongly identified as "model" without the POS information (Vlachidis and Tudhope, 2016). This study utilizes NLTK toolkits to perform POS tagging and lemmatization (Bird and Loper, 2004). Moreover, stop-words (i.e., a, an, of, one, two, three and so on) are removed, because they are non-descriptive and do not convey any semantic meanings.

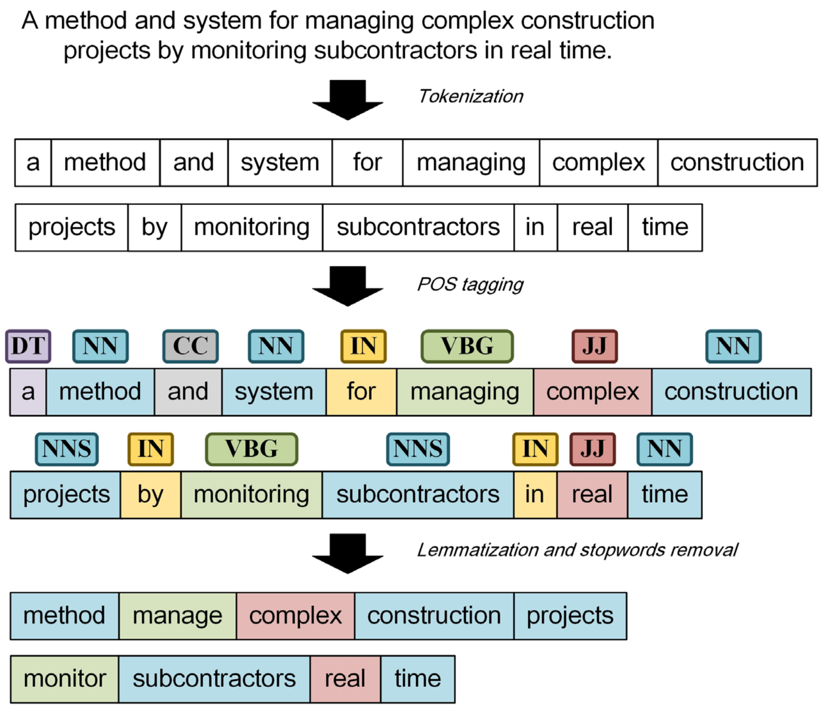


Figure 3 The processing procedure for textual data in patents

3.3. Vectorizing patents

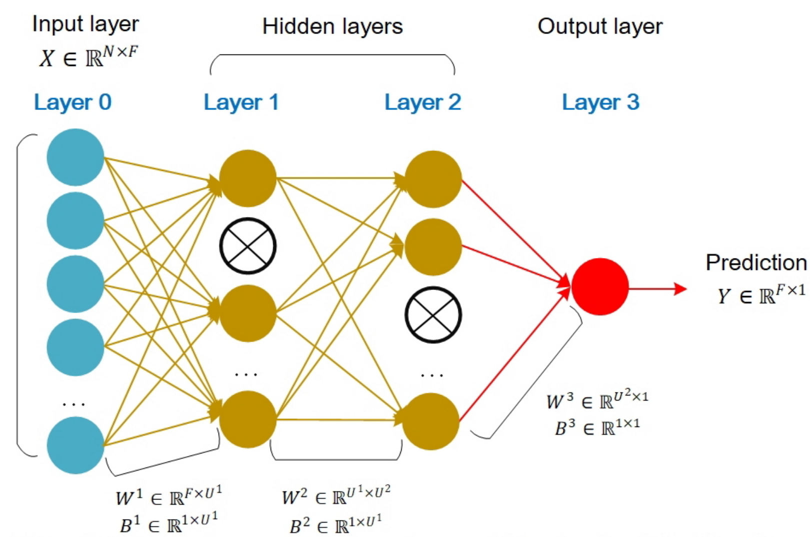
The processed patent texts have to be converted into numerical vectors that can be fed into MLP. This study adopts N-gram model with Tf-Idf weighting algorithm to vectorize the patents. N-gram considers the N words in a sequence as a feature, which has been proposed in the 1940s (Shannon, 1948) and has been employed in a large and growing body of literature (Bengio et al., 2003; Benson and Magee, 2013). In this case, two typical N-gram models, N = 1 (unigram) and 2 (bigram) are used to extract unigrams and bigrams as features from the patent texts, constituting the vocabulary with size v (overall v unigrams and bigrams are identified from the patent texts). A vector with v-dimension in which each position is the Tf-Idf (term frequency & inverse document frequency, see Sparck Jones (1972) for details) value of the feature in the vocabulary is generated to represent a patent. Another necessary step is to filter useful features because many of the features do not contribute to the training and prediction. This study, according to the Tf-Idf vectors, uses ANOVA F-value to select top features (number = K). In this study, K is set as a hyperparameter that would be tuned in the optimization step.

This study adopts N-gram and Tf-Idf as the vectorizing approach but not the topic models or SAO structures, because (1) BOW and Tf-Idf have been widely used in NLP studies and have been proven as the prominent vectorizing algorithm due to the simplicity and effectiveness (García Adeva et al., 2014; Mirończuk and Protasiewicz, 2018; Pavlinek and Podgorelec, 2017); (2) Topic models and SAO structures are suitable in clustering or classification tasks that have more than two classes to be distinguished (Blei et al., 2003; Choi et al., 2012); (3) Topic models and SAO structures replace the N-grams with latent topics or subject-active-objective structures. This would generate vectors with much lower dimensions which is not necessary in this case, because the proposed MLP model may get better performance when the number of input features is large (Cakir and Yilmaz, 2014).

3.4. MLP architecture

To improve the performance and take the non-linear relations into consideration, this study proposes MLP to learn and train the complex relations between inputs and outputs. MLP is typically designed with a feed-forward-based architecture and back-propagation learning process (Rosenblatt, 1961). There is a number of neurons in the MLP, and each of them receives signals from the former layer and pass transformed signals by an activation function to the subsequent layer (Riedmiller, 1994). Although it is a general wisdom that deep learning models are better than machine learning models, neural network design and hyperparameter choice are more important than deep learning models themselves (Levy et al., 2015). This section describes the MLP architecture.

After the vectorizing, the input matrix in this study is $X \in \mathbb{R}^{N\times F}$, in which N and F represent the number of instances and features respectively. Features are set as columns, and thus each patent is reflected as a row vector $x_i \in \mathbb{R}^{1\times F}$. The output is a column vector $Y \in \mathbb{R}^{N\times 1}$. The main target of the MLP model is to obtain learned neurons in layers of neural networks that could predict from X to Y. Figure 4 illustrates the architecture of the MLP consisting of four layers: one input layer, two hidden layers and an output layer, labeled from layer 0 to layer 3. The weigh matrixes connect the layers in sequence, and the neurons in the hidden and output layers are processing units, embodied with activation functions to transform the inputs to outputs. The number of the neurons in the input and output layers are set as N and 1, which are subject to the dimensions of the input and output vectors. The numbers of the neurons in the hidden layers are set as hyperparameters.



Notes: (1) The features are set as columns in the input matrix, therefore each neuron in the input layer is a F (the number of features) dimension vector, representing a patent.(2) ⊗ denotes the dropouts of the hidden layers in the training.

Figure 4 The MLP architecture

The MLP predicts the outputs based on the connection weights and the activation functions. In specific, the j-th neuron in l-th layer transforms an output based on the following equations:

$$\begin{cases} h_i^l = f^l\left(\sum_{i=1}^{U^{l-1}} h_i^{l-1} w_{ij}^l + b^l\right), l = 1,2 \\ h^l = h^3 = f^3\left(\sum_{i=1}^{U^2}(h_i^2 w_i^3 + b_i^3)\right), l = 3 \end{cases} \tag{1}$$

where l represents the layer sequence, U^{l-1} indicates the number of neurons in the $(l-1)$-th layer, x_i^{l-1} denotes the output of i-th neuron it receives, w_{ij} is the weight connecting x_i^{l-1} and j-th neuron in l-th layer, and b is the bias function for this neuron. f^l is the activation function in l-th layer. In this case, the two hidden layers (layer 1 and layer 2) and the output layer (layer 3) use *Rectified Linear Unit* (ReLU) and *Sigmoid* functions as the activation functions respectively. With the back-propagation process, the neurons in hidden and output layers can be trained with unique weight matrix and bias, producing different outputs according to the tasks (Garcia-Laencina et al., 2013). Moreover, the "Dropouts" is adopted in the hidden layers to prevent the overfitting (Srivastava et al., 2014).

3.5. Model training by gradients and dropouts

As mentioned above, the main task of MLP is to make the neurons to be learned, which could predict Y from X. The learning process is achieved by certain iterations, each of which is a loop consisting of a feed-forward and a back-propagation process (Haykin, 1999; Riedmiller, 1994). In the feed-forward process, the weights and bias in the hidden and output layers are randomly generated and propelled forward, calculating the output value h^3 from input X. Since a sigmoid function is selected as the activation function in the output layer, the errors follow a logistic distribution between the predictions (with values between 0 and 1) and true labels (with values are only 0 or 1). The loss function is the following:

$$J = -\sum_{n=1}^{N} y_n \log(h_n^3) + (1 - y_n) \log(1 - h_n^3) \quad (2)$$

In the back-propagation, the parameters θ (including all the weights and bias in hidden and output layers) would be updated by stochastic gradient descent. Two types of signals constitute the gradients: (1) global signals that can be computed from the derivatives, which transform the errors from the loss function; (2) local signals that are the inputs from the former layer. The θ would be updated from back to front, as the gradients are computed from the loss value to the former layers, one by one. For the clarity of the back-propagation process, this study illustrates the updating process of w_{ij}^2 and w_j^3 in layer 2 and 3. Figure 5 shows the functions of the neurons in layer 2 and 3. The gradient of w_j^3 (∇w_j^3) is defined as the derivative from J to w_j^3, which could be computed by the chain rule of derivatives:

$$\nabla w_j^3 = \frac{dJ}{dw_j^3} = \left(\frac{dJ}{dh^3} \times \frac{dh^3}{dN^3}\right) \times \frac{dN^3}{dw_j^3} = f'(N^3) \times h_j^2 \tag{3}$$

$$w_j^3 new = f(w_j^3 old, \nabla w_j^3) \tag{4}$$

where the $f'(N^3)$ is the global signal that could be computed by the derivative with loss value, h_j^2 is the local signal (the output of the j-th neuron in layer 2), and a is the learning rate that is pre-defined.

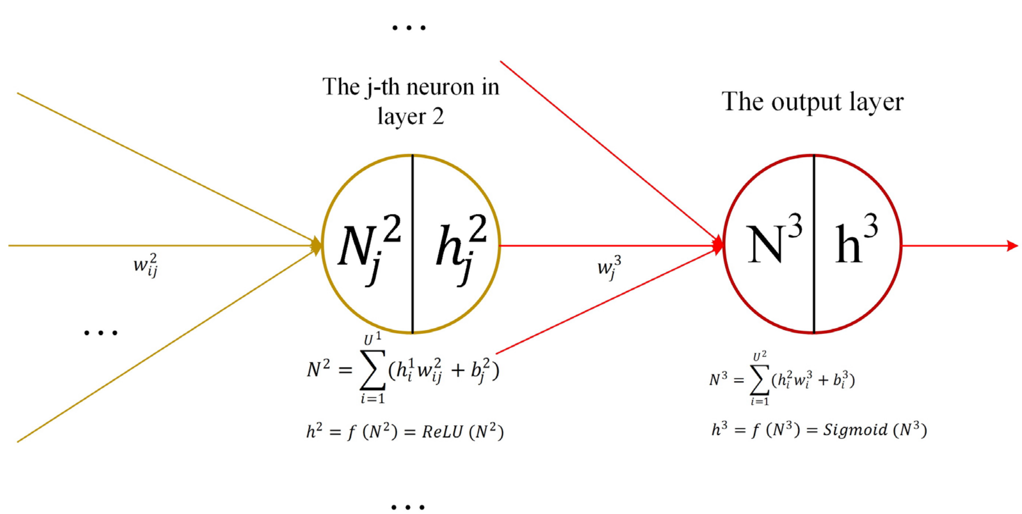


Figure 5 The neurons with input and activation functions in the last two layers

Similar to layer 3, ∇w_{ij}^2 could be computed by the following:

$$\nabla w_{ij}^2 = \frac{dJ}{dw_j^3} = \frac{dJ}{dh^2} \times \frac{dh^2}{dN^2} \times \frac{dN^2}{dw_{ij}^2} = f'(N^2) w_j^3 f'(N^3) \times h_i^1 \tag{5}$$

$$w_{ij}^2 new = f(w_{ij}^2 old, \nabla w_{ij}^2) \quad (6)$$

where $f'(N^2)w_j^3 f'(N^3)$ is the global signal that is propagated from the loss value, and h_i^1 is the local signal. The computations of other parameters, such as w and b are similar to equation (3) and (5). According to the gradients, the parameters could be updated by algorithms (equation (4) and (6)). Typical optimization algorithms include Stochastic Gradient descent (Robbins and Monro, 1985), AdaGrad (Duchi et al., 2011), RMSProp (Tieleman and Hinton, 2012), and Adam (Kingma and Ba, 2014). This study applies the Adam algorithm as the optimizer, as it has been recognized as the most effective in most cases with less computation time.

Dropouts is applied in the training process. “Dropouts” refers to temporarily eliminating some neurons and their incoming and outgoing connections in the neural networks. The dropped neurons are selected randomly based on a pre-defined ratio a (a=0.2 in this case). In the back-propagation of a training loop, a new thinned neural network is achieved with the proportion of 1 - a neurons remained. The parameters updating process would be implemented within the thinned neural network. In the feed-forward process of the subsequent loop, the removed neurons would turn on, and their parameters are obtained from the remaining neurons by a scale of 1/a. which parameters are obtained from the remaining neurons by a scale of 1/a. Therefore, training MLP with Dropouts can be regarded as training a larger number of thinned neural networks that share the same parameters. Such a training fashion effectively prevents neurons from co-adapting, and thus preventing the overfitting issues (Al Rahhal et al., 2018). As for the details of Dropouts, please see Al Rahhal et al. (2018).

After updating θ, a loop with a feed-forward and back-propagation finishes, which would be iterated in training. In this case, the maximum of epochs is set as 1000, and the consecutive tries of loss value without decrease is set two. The training process would iterate the loops until any of the above stop conditions is met. A small number of self-developed python programs are used to build, train, optimize and validate the model.

4. Results

4.1. Results of hyperparameters tuning

The purpose of hyperparameters tuning is to achieve an MLP model with the best performance by tuned hyperparameters. The range of the features is from 1000 to 40000, with steps of 1000 and 2500 for F in (1000, 10000) and (10000, 40000) respectively. With regard to the number of units, this study adopts the measurement proposed by Fan et al. (2015), which proposed a range around $\sqrt{N+1}$ (N denotes the number of neurons in the input layer). The resulting range of the number of units is from 5 to 69 and the step is set as 8. Figure 6 shows the hyperparameters tuning process. The MLP model reaches the highest accuracy when F is 30000 and U is 13.

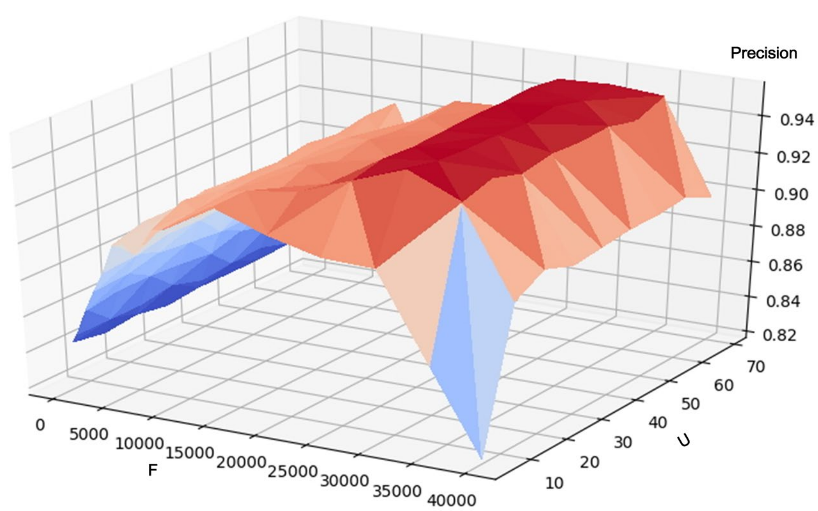


Figure 6 The hyperparameters tuning process

4.2. Validation

4.2.1. Validation methods

This study validates the proposed approach not only over the dataset in which 348 patents (labeled as *ICTC* or non-*ICTC*) were collected from USPTO, but also the patents from Derwent Innovations Index (DIX). The additional validation over DIX patents can evaluate the performance of the proposed model in processing texts that were written in different genres. Following prior machine learning studies (Sokolova and Lapalme, 2009), this study utilizes precision, recall, F-score to validate the deep learning model based on true positives (TP), false positives (FP) and false negatives (FN). Generally, TP is the number of instances the model correctly predicted. FP denotes the instances the model incorrectly predicted. FN reflects the number of instances the model failed to predict. The precision, recall, and F-score can be computed by

$$\mathrm{P} = \frac{TP}{TP+FP}, \mathrm{R} = \frac{TP}{TP+FN}, \mathrm{F1} = \frac{2\times P\times R}{P+R} \tag{7}$$

Specifically, as for the 348 USPTO patents in the dataset, we used the k-fold cross-validation along with the training process. In the training process, all the dumping data would be randomly split into k folds with the same size, and one of them is set as test instances and others are used for training. Such a training process is performed in k times, each of which has a different fold for testing and a different composition of k-1 folds for training. The final validation value is the average of k validation values. In this way, k-fold cross-validation prevents the bias in data selection and ensures the measures of the performances with objections (Friedman et al., 2001). In this study, k is set as 5, and all the annotated data (totally 348 USPTO patents annotated as *ICTC* or *non-ICTC* class were obtained in Section 3.1) were randomly divided into 5 folds. For each training round, four folds consist of the training collection and the other fold consists of the testing collection.

Besides the annotated data, this study collected and randomly selected 200 patents from Derwent Innovations Index (DIX) as an additional testing dataset, where the patents were written by inventors from a much wider range of countries and agencies. The search strategy is the same as the retrieval from USPTO: topic = *construction project, project management, infrastructure project, civil engineering* and *transportation project*. As the DIX does not provide fields of claim and description, title and abstract were collected as raw data. Before validation, the raw data of the 200 patents have to be annotated, processed and vectorized by the same processes mentioned in *Section 3.1, 3.2 and 3.3*.

4.2.2. Validation results

In this validation, the goal is to verify if the MLP has better screening accuracy than the traditional machine learning models. The performance of the MLP is compared against existing machine learning models, including Gaussian Naive Bayes (GNB), SVM and Bernoulli Naive Bayes (BNB). Figure 7 shows the accuracy over the different feature numbers. The highest precision value for each model are marked above the lines. By examining the figure, we can verify that the MLP model is superior to those machine learning models over all the features except K = 1000 and K = 40000. We can also observe that the MLP model is more sensitive to the number features, with the highest standard deviation value (0.032) over the models. This is consistent with one of the major differences between deep learning and traditional machine learning models: the traditional machine learning models are not capable of adjusting the model complexity according to the inputs, whereas the deep learning could tune the structure (number of layers and neurons) that is most suitable for input features (Moraes et al., 2013).

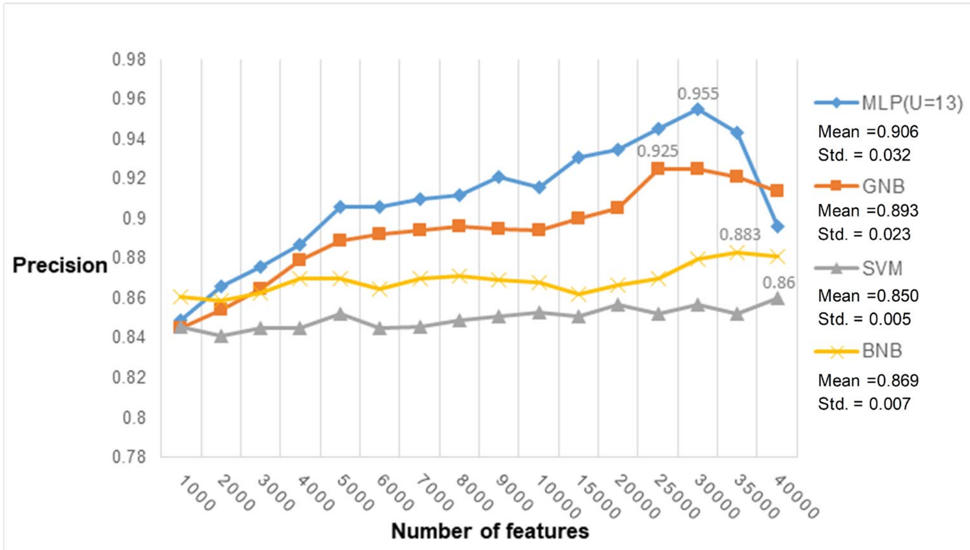


Figure 7 The precision values for MLP and machine learning models over the features

Table 3 illustrates the cross-validation results over the optimized MLP (K= 30000, U = 13), GNB SVM, and BNB. As was mentioned above, 5-fold cross-validation is used to verify the performance of the trained model. All the annotated instances were shuffled and randomly divided into 5 folds with same size, and 4 of them were used for training and the rest were testing instances. The training and testing would be preformed in 5 times, and each time has a different fold for testing and a different combination of 4 folds for training. The performance value is obtained by the mean of the 5 testing results. It can be observed that MLP (K= 30000, U = 13) has the best performance in all the three indexes (precision, recall and F1 score).

Table 3 Cross-validation results over MLP, GNB, SVM and BNB in the initial dataset

	Precision	**Recall**	**F1 score**
MLP (K=30000,U=13)	**0.955**	**0.954**	**0.954**
GNB (K=25000)	0.925	0.919	0.918
SVM (K=40000)	0.86	0.852	0.848
BNB (K=35000)	0.883	0.86	0.853

Table 4 shows the validation results over another database, which is used to evaluate the generality of the proposed model. The validation results, described by Table 4, indicate that the learned classifier based on MLP could be precisely implemented in the database from DIX, in which patents are written in different levels by a variety of inventors from different countries. In addition, the MLP also outperforms the machine learning methods.

Table 4 Cross-validation results over MLP, GNB, SVM and BNB in the dataset from DIX

	Precision	**Recall**	**F1 score**
MLP (K=30000,U=13)	**0.897**	**0.897**	**0.897**
GNB (K=25000)	0.849	0.852	0.849
SVM (K=40000)	0.85	0.852	0.848
BNB (K=35000)	0.444	0.667	0.533

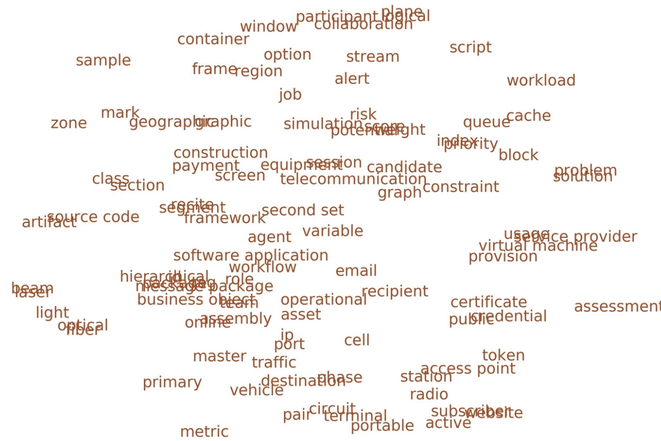
4.3. The screened ICTC patents

Besides the validation results, some important implications should be further discussed. The authors use the proposed approach to automatically screen a corpus of ICTC patents. To compare the topic distribution of the patents in the corpus, as well as the patents in collection 1 and 2 (Table 1), this study plots the figures of feature space for each of the collections (Figure 8). According to the processes in section 3.1 - 3.3, this study vectorizes each of the patents in the three collections. The t-Distributed Stochastic Neighbor Embedding (TSNE) algorithm (Czerniawski et al., 2018; Maaten and Hinton, 2008) is adopted to project the high dimensional feature vectors into a 2D plot, in which the physical distance between two features roughly represents the degree of association of them in the corresponding collection.

(a)

Number of patents: 5311

No.	Term	No.	Term
1	Asset	16	Score
2	Virtual machine	17	Class
3	Package	18	Light
4	Recite	19	Optical
5	Station	20	Terminal
6	Workflow	21	Circuit
7	Agent	22	Block
8	Risk	23	Participant
9	Session	24	Stream
10	Logical	25	Subscriber
11	Metric	26	Phase
12	Construction	27	Business object
13	Vehicle	28	Role
14	Tag	29	Certificate
15	Segment	30	Section



(b)

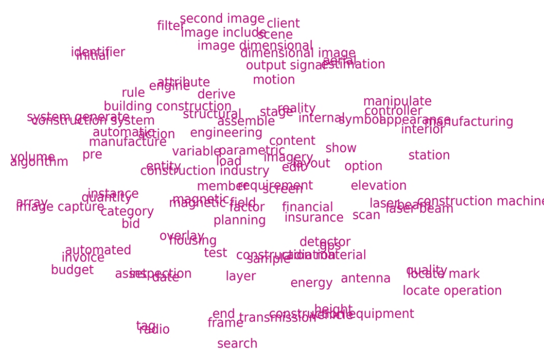
Number of patents: 922

No.	Term	No.	Term
1	Enforcer	16	Embodiment IP
2	Policy enforcer	17	Process failure
3	IP asset	18	IP mate
4	IP marketplace	19	Lane
5	Policy abstraction	20	Return signal
6	Reporting source	21	Policy engine
7	Report source	22	Hosting
8	Policy server	23	Web host
9	WAF	24	Parametric
10	Event entry	25	Secure exchange
11	Addressable interface	26	Actuation system
12	LUW	27	Actuation set
13	Block least	28	Software artifact
14	Information asset	29	Point cloud
15	GUI component	30	Policy language

(c)

Number of patents: 1818

No.	Term	No.	Term
1	laser	16	radio
2	antenna	17	reality
3	parametric	18	bid
4	asset	19	quality
5	construction machine	20	scan
6	structural	21	stage
7	aerial	22	beam
8	vehicle	23	layer
9	layout	24	inspection
10	construction equipmen	25	search
11	construction material	26	end
12	magnetic	27	action
13	controller	28	rule
14	symbol	29	radiation
15	tag	30	transmission



Notes: (2) In each sub-figure, top 100 features with highest average Tf-Idf value are plotted, and top 30 are list at left for clarity. (2) As for strategy 1 and 2, please see Table 6.2 for details.

Figure 8 TSNE plots of feature spaces (a) The plot of patents screened by strategy 1; (b) The plot of patents screened by strategy 2; (c) The plot of patents screened by the proposed approach

As explained in the introduction, the searching engines for patents has two major flaws: (1) the searching engines can only perform "match" logic based on structured data; and (2) searching by keywords cannot avoid personal preference, and thus the results highly depend on users' knowledge. Figure 8 (a) depicts the feature space of collection 1, in which patents were searched by ICT classes and AEC domain keywords. The features in this figure are averagely distributed, incorporating a large number of ICT-related features, but some typical ICTC terms do not appear. Such feature distribution indicates that the patents in this collection are mainly relevant to ICT, but not ICTC. A possible explanation for this might be that the AEC domain keywords are not capable of discerning ICTC patents from ICT patents using query-based methods. For example, the keyword "construction project" may match patents related to construction projects, but it can also match patents of "software project" containing sayings about "construction project" which means construct a project. Despite the miss matching problem, strategy 2 can lead to short coverage of ICT techniques. As Figure 8 (b) shown, the features are agglomerated into clusters, indicating an unbalanced distribution of topics. The features, not surprised, are mainly related to the searching keywords, such as wireless and mobile. The features in Figure 8 (C) are distributed averagely, incorporating a wide range of ICTC related terminologies, such as laser, construction machine, and radio. This indicates that the proposed approach is more suitable in gathering ICTC patents than traditional searching engines.

5. Discussion and conclusion

ICT applications are a key determinant to improve the level of coordination and collaboration in the AEC industry. Even though patents have been recognized as a valuable resource to provide technological knowledge, the patent offices have not provided a specific classification of ICTC patents. Acknowledging this research opportunity, the presenting study accurately and widely screens a corpus of ICTC patents, by proposing an approach based on deep learning and NLP techniques.

Specifically, this study has made the following contributions: (1) This study contributes an approach to widely and accurately retrieve and collect a corpus of patents for domains like ICTC that does not exist as a specific classification in patents and hardly being represented by queries. Although patent offices provide elaborate classification schemes, it cannot satisfy all the requirements in the real world. Therefore, when a collection of patents does not exist in the classification schemes, query-based methods become the only possible way to search these patents. However, query-based methods were developed for retrieving relevant documents for a specific patent application rather than a set of patents. For the collections like ICTC that incorporates a variety of technologies, it is an extremely challenging task for the query-based methods to retrieve the patents simply by a query. (2) The proposed approach takse advantage of deep learning and NLP techniques. Although deep learning has become prominent in processing textual data, the previous studies in the AEC area mainly utilized machine learning methods to perform classification tasks, which performance highly depends on feature selection because the traditional machine learning models could only learn the linear relations. Compared to traditional machine learning methods, deep learning models are more advanced by using the layers of neural networks to learn non-linear relations and more suitable for complex tasks with specific objectives. The validation results indicate that the MLP model outputs the traditional machine models in classifying the ICTC patents. In addition, NLP techniques were employed to pre-process the raw data. In the AEC area, most previous studies only utilized the N-gram, tokenization and stop-words removal, ignoring the advanced NLP tools such as lemmatization and POS. This study utilizes lemmatization and POS to convert the words into stems for generating more accurate N-grams from the textual data. (3) In practice, this study contributes a specific collection for ICTC patents, which is not provided by the patent offices. The collection widely and accurately covers the ICT applications in the construction, not only constituting a dictionary for searching ICTC, but also identifying problems to be solved by the state of art ICTC inventions and recognizing all possible specific embodiments of ICTC.

The presenting study is not without limitations. The feature extraction process is based on traditional BOW models, which does not take the semantic meanings in contexts into consideration. This limitation, however, has been largely offset by the proposed supervised MLP model, which learns complex relations between the inputs and outputs by training the deep layers of neurons. This could perform the prediction task with good performance without considering the semantic meanings. This study focusses on classifying ICTC and non-ICTC. Future research is needed to concentrate on AI-aided approaches that could automatically categorize the ICTC patents according to the technological components or the management issues in practice.

References

Adriaanse, A., Voordijk, H., Dewulf, G., 2010. The use of interorganisational ICT in United States construction projects. Automation in Construction, 19(1), 73-83.

Aggarwal, C.C., Reddy, C.K., 2013. Data clustering: algorithms and applications. CRC press.

Agrawal, A., Henderson, R., 2002. Putting patents in context: Exploring knowledge transfer from MIT. Management Science, 48(1), 44-60.

Ahuja, V., Yang, J., Shankar, R., 2009. Study of ICT adoption for building project management in the Indian construction industry. Automation in Construction, 18(4), 415-423.

Ahuja, V., Yang, J., Skitmore, M., Shankar, R., 2010. An empirical test of causal relationships of factors affecting ICT adoption for building project management: An Indian SME case study. Construction innovation, 10(2), 164-180.

Al Rahhal, M.M., Bazi, Y., Al Zuair, M., Othman, E., BenJdira, B., 2018. Convolutional Neural Networks for Electrocardiogram Classification. Journal of Medical and Biological Engineering, 38(6), 1014-1025.

Alberts, D., Yang, C.B., Fobare-DePonio, D., Koubek, K., Robins, S., Rodgers, M., Simmons, E., DeMarco, D., 2017. Introduction to patent searching, Current Challenges in Patent

Information Retrieval. Springer, pp. 3-45.

Alsafouri, S., Ayer, S.K., 2018. Review of ICT Implementations for Facilitating Information Flow between Virtual Models and Construction Project Sites. Automation in Construction, 86(August 2016), 176-189.

Azad, H.K., Deepak, A., 2019. Query expansion techniques for information retrieval: A survey. Information Processing & Management, 56(5), 1698-1735.

Bell, G., Hey, T., Szalay, A., 2009. Beyond the data deluge. Science, 323(5919), 1297-1298.

Bengio, Y., Ducharme, R., Vincent, P., Jauvin, C., 2003. A neural probabilistic language model. Journal of Machine Learning Research, 3(Feb), 1137-1155.

Benson, C.L., Magee, C.L., 2013. A hybrid keyword and patent class methodology for selecting relevant sets of patents for a technological field. Scientometrics, 96(1), 69-82.

Bird, S., Loper, E., 2004. NLTK: the natural language toolkit, Proceedings of the ACL 2004 on Interactive poster and demonstration sessions. Association for Computational Linguistics, p. 31.

Blei, D.M., Ng, A.Y., Jordan, M.I., 2003. Latent dirichlet allocation. Journal of Machine Learning Research, 3, 993-1022.

Bouadjenek, M.R., Sanner, S., Ferraro, G., 2015. A study of query reformulation for patent prior art search with partial patent applications, Proceedings of the 15th International Conference on Artificial Intelligence and Law. ACM, pp. 23-32.

Brown, I., Mues, C., 2012. An experimental comparison of classification algorithms for imbalanced credit scoring data sets. Expert Systems with Applications, 39(3), 3446-3453.

Cakir, L., Yilmaz, N., 2014. Polynomials, radial basis functions and multilayer perceptron neural network methods in local geoid determination with GPS/levelling. Measurement, 57, 148-153.

Cambria, E., White, B., 2014. Jumping NLP curves: A review of natural language processing research. IEEE Computational Intelligence Magazine, 9(2), 48-57.

Cassetta, E., Marra, A., Pozzi, C., Antonelli, P., 2017. Emerging technological trajectories and new mobility solutions. A large-scale investigation on transport-related innovative start-ups and implications for policy. Transportation Research Part A: Policy and Practice, 106(March), 1-11.

Chakrabarti, S., Dom, B., Indyk, P., 1998. Enhanced hypertext categorization using hyperlinks. SIGMOD Rec., 27(2), 307-318.

Chiarello, F., Cimino, A., Fantoni, G., Dell'Orletta, F., 2018. Automatic users extraction from patents. World Patent Information, 54, 28-38.

Choi, S., Kang, D., Lim, J., Kim, K., 2012. A fact-oriented ontological approach to SAO-based function modeling of patents for implementing Function-based Technology Database. Expert Systems with Applications, 39(10), 9129-9140.

Cocarascu, O., Toni, F., 2018. Combining Deep Learning and Argumentative Reasoning for the Analysis of Social Media Textual Content Using Small Data Sets. Computational Linguistics, 44(4), 833-858.

Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., Kuksa, P., 2011. Natural language processing (almost) from scratch. Journal of Machine Learning Research, 12(Aug), 2493-2537.

Czerniawski, T., Sankaran, B., Nahangi, M., Haas, C., Leite, F., 2018. 6D DBSCAN-based segmentation of building point clouds for planar object classification. Automation in Construction, 88, 44-58.

Davies, R., Harty, C., 2013. Implementing 'Site BIM': A case study of ICT innovation on a large hospital project. Automation in Construction, 30, 15-24.

Dehlin, S., Olofsson, T., 2008. An evaluation model for ICT investments in construction projects. Electronic journal of information technology in construction, 13, 343-361.

Duchi, J., Hazan, E., Singer, Y., 2011. Adaptive subgradient methods for online learning and stochastic optimization. Journal of Machine Learning Research, 12(Jul), 2121-2159.

El-Ghandour, W., Al-Hussein, M., 2004. Survey of information technology applications in construction. Construction innovation, 4(2), 83-98.

Enesi, F.A., Oyefolahan, I.O., Abdullahi, M.B., Salaudeen, M.T., 2018. Enhanced Query Expansion Algorithm: Framework for Effective Ontology Based Information Retrieval System. i-Manager's Journal on Computer Science, 6(4), 1.

Fall, C., Benzineb, K., Guyot, J., Törcsvári, A., Fiévet, P., 2003a. Computer-assisted categorization of patent documents in the international patent classification, Proceedings of the International Chemical Information Conference (ICIC'03), Nîmes, France.

Fall, C.J., T, A., #246, rcsv, #225, ri, Benzineb, K., Karetka, G., 2003b. Automated categorization in the international patent classification. SIGIR Forum, 37(1), 10-25.

Fan, X., Li, S., Tian, L., 2015. Chaotic characteristic identification for carbon price and an multi-layer perceptron network prediction model. Expert Systems with Applications, 42(8), 3945-3952.

Flyvbjerg, B., 2014. What You Should Know About Megaprojects and Why: An Overview. Project Management Journal, 45(2), 6-19.

Forman, G., 2002. Choose Your Words Carefully: An Empirical Study of Feature Selection Metrics for Text Classification. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 150-162.

Friedman, J., Hastie, T., Tibshirani, R., 2001. The elements of statistical learning. Springer series in statistics New York, NY, USA:.

Frits, S., 2007. Strategy to enhance use of ICT in construction, Proceedings of CIB World Building Congress, pp. 2527-2535.

Garcia-Laencina, P.J., Sancho-Gomez, J.L., Figueiras-Vidal, A.R., 2013. Classifying patterns with missing values using Multi-Task Learning perceptrons. Expert Systems with Applications, 40(4), 1333-1341.

García Adeva, J.J., Pikatza Atxa, J.M., Ubeda Carrillo, M., Ansuategi Zengotitabengoa, E., 2014. Automatic text classification to support systematic reviews in medicine. Expert Systems with Applications, 41(4 PART 1), 1498-1508.

Gerken, J.M., 2012. A new instrument for technology monitoring: novelty in patents measured by semantic patent analysis. Scientometrics, 91(3), 645-670.

Giachanou, A., Salampasis, M., Paltoglou, G., 2015. Multilayer source selection as a tool for supporting patent search and classification. Information Retrieval Journal, 18(6), 559-585.

Gimpel, K., Schneider, N., O'Connor, B., Das, D., Mills, D., Eisenstein, J., Heilman, M., Yogatama, D., Flanigan, J., Smith, N.A., 2011. Part-of-Speech Tagging for Twitter: Annotation, Features, and Experiments, Meeting of the Association for Computational Linguistics: Human Language Technologies: Short Papers, pp. 42-47.

Girthana, K., Swamynathan, S., 2018. Semantic Query-Based Patent Summarization System (SQPSS), International Conference on Intelligent Information Technologies. Springer, pp. 169-179.

Greiman, V.A., 2013. Megaproject management: Lessons on risk and project management from the Big Dig. John Wiley & Sons.

Gwak, J.H., Sohn, S.Y., 2018. A novel approach to explore patent development paths for subfield technologies. Journal of the Association for Information Science and Technology, 69(3), 410-419.

Habash, N., Rambow, O., Roth, R., 2009. MADA+ TOKAN: A toolkit for Arabic tokenization, diacritization, morphological disambiguation, POS tagging, stemming and lemmatization, Proceedings of the 2nd international conference on Arabic language resources and tools (MEDAR), Cairo, Egypt, p. 62.

Haddi, E., Liu, X., Shi, Y., 2013. The role of text pre-processing in sentiment analysis. Procedia Computer Science, 17, 26-32.

Haykin, S., 1999. Neural networks a comprehensive introduction. Prentice Hall, New Jersey.

Hoetker, G., Agarwal, R., 2007. Death Hurts, But It Isn't Fatal: The Postexit Diffusion of Knowledge Created by Innovative Companies. Academy of Management Journal, 50(2), 446-467.

Intarakumnerd, P., Charoenporn, P., 2015. Impact of stronger patent regimes on technology transfer: The case study of Thai automotive industry. Research Policy, 44(7), 1314-1326.

Ittoo, A., Nguyen, L.M., van den Bosch, A., 2016. Text analytics in industry: Challenges, desiderata and trends. Computers in Industry, 78, 96-107.

Jain, A., Kulkarni, G., Shah, V., 2018. Natural Language Processing. International Journal of Computer Sciences and Engineering, 6(1), 161-167.

Kaplan, S., Vakili, K., 2013. Novelty vs. usefulness in innovative breakthroughs: A test using topic modeling of nanotechnology patents, Technical Report, Working Paper.

Kingma, D.P., Ba, J., 2014. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.

Kurdi, M.Z., 2017. Natural language processing and computational linguistics 2: semantics, discourse and applications. John Wiley & Sons.

Lee, C., Song, B., Park, Y., 2013. How to assess patent infringement risks: a semantic patent claim analysis using dependency relationships. Technology Analysis and Strategic Management, 25(1), 23-38.

Levitt, R.E., 2007. CEM research for the next 50 years: Maximizing economic, environmental, and societal value of the built environment. Journal of Construction Engineering and Management-Asce, 133(9), 619-628.

Levy, O., Goldberg, Y., Dagan, I., 2015. Improving distributional similarity with lessons learned from word embeddings. Transactions of the Association for Computational Linguistics, 3, 211-225.

Li, H., Chan, G., Wong, J.K.W., Skitmore, M., 2016. Real-time locating systems applications in construction. Automation in Construction, 63, 37-47.

Li, S., Hu, J., Cui, Y., Hu, J., 2018. DeepPatent: patent classification with convolutional neural networks and word embedding. Scientometrics, 117(2), 721-744.

Li, X., Shen, G.Q., Wu, P., Yue, T., 2019. Integrating Building Information Modeling and Prefabrication Housing Production. Automation in Construction, 100, 46-60.

Li, Z., Tate, D., Lane, C., Adams, C., 2012. 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), 987-1010.

Liu, S.-H., Liao, H.-L., Pi, S.-M., Hu, J.-W., 2011. Development of a Patent Retrieval and Analysis Platform – A hybrid approach. Expert Systems with Applications, 38(6), 7864-7868.

Maaten, L., Hinton, G., 2008. Visualizing data using t-SNE. Journal of Machine Learning Research,

9(2605), 2579-2605.
Mahdabi, P., Crestani, F., 2014. The effect of citation analysis on query expansion for patent retrieval. Information Retrieval, 17(5-6), 412-429.
Mahdabi, P., Keikha, M., Gerani, S., Landoni, M., Crestani, F., 2011. Building Queries for Prior-Art Search, Information Retrieval Facility Conference.
Michel, J., Bettels, B., 2001. Patent citation analysis.A closer look at the basic input data from patent search reports. Scientometrics, 51(1), 185-201.
Mirończuk, M.M., Protasiewicz, J., 2018. A recent overview of the state-of-the-art elements of text classification. Expert Systems with Applications, 106, 36-54.
Mok, K.Y., Shen, G.Q., Yang, J., 2015. Stakeholder management studies in mega construction projects: A review and future directions. International Journal of Project Management, 33(2), 446-457.
Moraes, R., Valiati, J.F., Neto, W.P.G., 2013. Document-level sentiment classification: An empirical comparison between SVM and ANN. Expert Systems with Applications, 40(2), 621-633.
Munková, D., Munk, M., Vozár, M., 2013. Data pre-processing evaluation for text mining: transaction/sequence model. Procedia Computer Science, 18, 1198-1207.
Nadeau, D., Sekine, S., 2007. A survey of named entity recognition and classification. Lingvisticae Investigationes, 30(1), 3-26.
Niemann, H., Moehrle, M.G., Frischkorn, J., 2017. 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, 210-220.
Onan, A., Korukoğlu, S., Bulut, H., 2016. Ensemble of keyword extraction methods and classifiers in text classification. Expert Systems with Applications, 57, 232-247.
Pavlinek, M., Podgorelec, V., 2017. Text classification method based on self-training and LDA topic models. Expert Systems with Applications, 80, 83-93.
Perez-Molina, E., 2018. The Role of Patent Citations as a Footprint of Technology. Journal of the Association for Information Science and Technology, 69(4), 610-618.
Riedmiller, M., 1994. Advanced supervised learning in multi-layer perceptrons-from backpropagation to adaptive learning algorithms. Computer standards and interfaces, 16(3), 265-278.
Robbins, H., Monro, S., 1985. A stochastic approximation method, Herbert Robbins Selected Papers. Springer, pp. 102-109.
Rosenblatt, F., 1961. Principles of neurodynamics. perceptrons and the theory of brain mechanisms. CORNELL AERONAUTICAL LAB INC BUFFALO NY.
Saiki, T., Akano, Y., Watanabe, C., Tou, Y., 2006. A new dimension of potential resources in innovation: A wider scope of patent claims can lead to new functionality development. Technovation, 26(7), 796-806.
Sardroud, J.M., 2015. Perceptions of automated data collection technology use in the construction industry. Journal of Civil Engineering and Management, 21(1), 54-66.
Shalaby, W., Zadrozny, W., 2018. Toward an interactive patent retrieval framework based on distributed representations, The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval. ACM, pp. 957-960.
Shalaby, W., Zadrozny, W., 2019. Patent retrieval: a literature review. Knowledge and Information systems, 1-30.
Shannon, C.E., 1948. A mathematical theory of communication. Bell system technical journal,

27(3), 379-423.

Shekarpour, S., Marx, E., Ngonga Ngomo, A.C., Auer, S., 2015. SINA: Semantic interpretation of user queries for question answering on interlinked data. Journal of Web Semantics, 30, 39-51.

Silva, F.N., Amancio, D.R., Bardosova, M., Costa, L.D., Oliveira, O.N., 2016. Using network science and text analytics to produce surveys in a scientific topic. Journal of Informetrics, 10(2), 487-502.

Silver, D., Huang, A., Maddison, C.J., Guez, A., Sifre, L., Van Den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M., 2016. Mastering the game of Go with deep neural networks and tree search. Nature, 529(7587), 484.

Smith, H., 2002. Automation of patent classification. World Patent Information, 24(4), 269-271.

Sokolova, M., Lapalme, G., 2009. A systematic analysis of performance measures for classification tasks. Information Processing & Management, 45(4), 427-437.

Sparck Jones, K., 1972. A statistical interpretation of term specificity and its application in retrieval. Journal of documentation, 28(1), 11-21.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R., 2014. Dropout: a simple way to prevent neural networks from overfitting. The Journal of Machine Learning Research, 15(1), 1929-1958.

Tannebaum, W., Rauber, A., 2014. Using query logs of USPTO patent examiners for automatic query expansion in patent searching. Information Retrieval, 17(5-6), 452-470.

Terragno, P.J., 1979. Patents as technical literature. IEEE Transactions on Professional Communication, PC-22(2), 101-104.

Tieleman, T., Hinton, G., 2012. Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. COURSERA: Neural networks for machine learning, 4(2), 26-31.

USPTO, 2007. Manual of patent examining procedure (8th ed.), Alexandria.

Venugopalan, S., Rai, V., 2015. Topic based classification and pattern identification in patents. Technological Forecasting and Social Change, 94, 236-250.

Vlachidis, A., Tudhope, D., 2016. A knowledge - based approach to I nformation E xtraction for semantic interoperability in the archaeology domain. Journal of the Association for Information Science and Technology, 67(5), 1138-1152.

Wang, J., 2018. Innovation and government intervention: A comparison of Singapore and Hong Kong. Research Policy, 47(2), 399-412.

Wang, X., Jiang, W., Luo, Z., 2016. 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, pp. 2428-2437.

Wu, C.H., Yun, K., Huang, T., 2010. Patent classification system using a new hybrid genetic algorithm support vector machine. Applied Soft Computing Journal, 10(4), 1164-1177.

Wu, P., Song, Y., Shou, W., Chi, H., Chong, H.-Y., Sutrisna, M., 2017. A comprehensive analysis of the credits obtained by LEED 2009 certified green buildings. Renewable and Sustainable Energy Reviews, 68, 370-379.

Zhang, L., Liu, Z., Li, L., Shen, C., Li, T., 2018. PatSearch: an integrated framework for patentability retrieval. Knowledge and Information systems, 57(1), 135-158.

Zhao, Z., Xu, S., Kang, B.H., Kabir, M.M.J., Liu, Y., Wasinger, R., 2015. Investigation and improvement of multi-layer perceptron neural networks for credit scoring. Expert Systems with Applications, 42(7), 3508-3516.

Zidane, Y.J.T., Johansen, A., Ekambaram, A., 2013. Megaprojects - Challenges and Lessons

Learned, in: Pantouvakis, J.P. (Ed.), Selected Papers from the 26th Ipma, pp. 349-357.