Automatic Review of Construction Specifications Using Natural Language Processing

Seonghyeon Moon[1]; Gitaek Lee[2]; Seokho Chi, Ph.D., M.ASCE[3]; and Hyunchul Oh, Ph.D.[4]

[1]Construction Innovation Laboratory, Dept. of Civil and Environmental Engineering, Seoul National Univ., Seoul 08826, South Korea. E-mail: blank54@snu.ac.kr
[2]Construction Innovation Laboratory, Dept. of Civil and Environmental Engineering, Seoul National Univ., Seoul 08826, South Korea. E-mail: lgt0427@snu.ac.kr
[3]Construction Innovation Laboratory, Dept. of Civil and Environmental Engineering, Seoul National Univ., Seoul 08826, South Korea. E-mail: shchi@snu.ac.kr
[4]Daewoo E&C, Smart Construction Team, Seoul 08826, South Korea. E-mail: hyunchul.oh@daewooenc.com

ABSTRACT

Since construction specifications are normally over 1000 pages and are complicated and often inconsistent, reviewing them is a labor-intensive and time-consuming activity. Thus, the aim of this study was to automate the review process by comparing construction specifications with standard specifications using natural language processing. Standard specifications for road construction projects were collected from 43 different states in the U.S. and used as experimental data. Doc2Vec, cosine similarity, and named entity recognition (NER) were used to recognize construction objects, standard values, and execution conditions, which can be used to find specification errors. As an early stage of the research, most of related sentences were found from standard specifications with high relevancy, and the average F1 score of NER was 0.256. The research findings will contribute to enhancing the efficiency of checking for specification errors by automatically detecting abnormalities and the absence of specific standards.

INTRODUCTION

Reviewing construction specifications is a crucial process for contractors because they must follow the clients' requirements, which are stated clearly in the document. Failure to meet the standards of the document causes economic, technical, and social problems. For example, when Korean contractors were working on the road construction site in Qatar, problems with the asphalt pavement occurred due to the use of incorrect construction standards. The contractors performed the construction according to the specifications provided by the client, but the design criteria specified in the specifications were not suitable for the environment of Qatar; especially the hot weather condition. As a result, the contractors had conflicts with the client and the designers, resulting in the waste of resources and delays in the project.

Even though reviewing the construction specifications are crucial, it is difficult to analyze the documents due to following issues. First, the documents are generally complicated and contained errors because some of the clients, if they do not have their own standardized specifications like Qatar, are not familiar with construction standards and just sometimes tend to piece together parts of other specifications without careful investigations. In addition, the reviewing process is time-consuming and expensive since it is performed manually. In addition, the specifications are interpreted inconsistently because the reviewers are often unfamiliar with the local situations (e.g., environment, technical skills, and regulations).

To summarize, manual reviews of construction specifications waste time, increase costs, and contain inconsistent interpretations. To address these problems, in this research, our aim was to develop an automatic process of reviewing construction specifications using Natural Language Processing (NLP). Since this research is in its early stages, the overall flow of the research is described in this paper, including (1) selection of comparable specifications, (2) identification of corresponding sentences, (3) extraction of construction standards, and (4) comparison of construction standards.

The construction specifications for the Qatar highway construction site in 2014 was used for the analysis, and the standard specifications for road construction in 43 states in the U.S. were used as the reference set. We collected from websites the most recent specifications for 43 states in the U.S. since they permitted the specifications to be downloaded. The main beneficiary of the research would be the construction companies whose employees should ascertain the appropriateness of the clauses in the construction specifications.

RELATED WORKS

Natural Language Processing and Text Mining

NLP is a research area that utilizes various machine learning algorithms to process readable text, which enables a computer to analyze the text data (Zhang and El-Gohary, 2016). Since there is a large volume of documents that pertain to the construction industry, many researchers have analyzed the data they contain to manage the empirical information that is available in the documents. The text data in the construction industry include, for instance, regulations, bidding documents, specifications, construction reports, accident reports, and claim documents.

Text mining is a research concept in which text data (i.e., unstructured data) are processed by NLP and then analyzed by computer to extract information and determine relationships between the sets of information (Lee et al., 2016). The field of text mining in construction covers visualization, automatic summarization, information retrieval, ontology development, compliance checking, and other categories.

FUNDAMENTAL RESEARCH METHODOLOGY

Preprocessing

The text data used in the research went through three steps of preprocessing to be converted into a clean and computer-understandable format. The preprocessing steps consisted of tokenization, stopword removal, and stemming.

First, in the tokenization step, the research separated the text into several tokens, a minimum unit of text analysis, such as a document, paragraph, sentence, and word. This process was to prepare the text for feature representation that would be essential in the following analysis. In general, the ‘word,’ a chunk of alphabetical characters divided by space marks, is the most common unit used to analyze text. In addition, in this research, the combination of a punctuation mark (e.g., ‘.’, ‘,’, and ‘!’) and a space mark (e.g., ‘ ’, and ‘\n’) was used as a delimiter in order to separate sentences.

Second, in the stopword removal step, words that appeared in the text too often and were not significantly important in the analysis of the text were eliminated. The eliminated words are called ‘stopwords,’ which include grammatical elements, such as definite and indefinite articles (e.g., ‘a’, ‘an’, and ‘the’), prepositions (e.g., ‘to’, ‘on’, ‘in’), and pronouns (e.g., ‘he’, ‘she’, ‘it’).

Third, in the stemming step, the words that remained after the stopwords were removed were pruned into root or stem forms to map the various forms of words that have the same meaning to one unique term. For example, ‘construct,’ ‘construction,’ ‘constructor,’ and ‘constructing’ would be pruned to one term, ‘constr.’ The stemming process shortened the computing time by reducing the size of the word feature matrix, and it enhanced the quality of the analytical results by representing various words with essentially the same meaning with one word.

Text Embedding

Text embedding is a kind of text representation method, mapping the text on a real number vector space, the purpose of which is to use the text vector as input data to machine learning models (Chopra et al., 2016). This process is essential in NLP to conduct language modeling and feature learning. While there are several methods for text embedding, in this research, we used Term Frequency & Inverse Document Frequency (TF-IDF), Word2Vec, and Doc2Vec. The details of each method are provided below.

TF-IDF conserves the frequency of text data, which implicates the appearance (i.e., whether or not the frequency is zero) and importance (i.e., how many times the text appears). Term Frequency (TF) indicates the number of occurrences of a term in a document, which implicates the frequency of a term. Inverse Document Frequency (IDF) indicates an inverse number of documents that contain a certain term. TF-IDF represents text data via the importance of consisting terms (i.e., TF) normalized by IDF. The TF-IDF is calculated as shown in Equations 1-3, where *t*, *d*, and *c* indicate term, document, and corpus, respectively, and $f(w,d)$ indicates the frequency with which term *t* appeared in document *d*.

$$\boldsymbol{TFIDF}(t,d,c) = \boldsymbol{TF}(t,d) \times \boldsymbol{IDF}(t,c) \qquad \text{Equation (1)}$$

$$\boldsymbol{TF}(t,d) = 0.5 + \frac{0.5 \times f(t,d)}{max\{f(w,d) : w \in d\}} \qquad \text{Equation (2)}$$

$$\boldsymbol{IDF}(t,c) = \log \frac{|c|}{|\{d \in c : t \in d\}|} \qquad \text{Equation (3)}$$

Word2Vec is a neural network language model to learn word vectors, which models word-to-word relationships (Mikolov et al., 2013). The word-to-word relationship means the distribution of surrounding words, which could implicate the usage pattern of each word. Technically, the objective function of Word2Vec is to maximize the log probability of a target word given its surrounding words, provided as Equation 4.

$$\log P(w_O \mid w_I) = \log \sigma\left(v'_{w_O} \cdot v_{w_I}\right) + \sum_{i=1}^{k} w_i \sim P_n(w)\left[\log \sigma\left(-v'_{w_i} \cdot v_{w_I}\right)\right] \qquad \text{Equation (4)}$$

where w_O is the target word (output word), w_I is one of the surrounding words (input word), σ is the sigmoid function, k is the number of negative samples, $P_n(w)$ is the noise distribution, v_w is the vector of word w, and v'_w is the negative sample vector.

Doc2Vec is an extended version of Word2Vec to represent longer text (e.g., sentence, paragraph, and document). The document vector would be generated according to the combination of the Word2Vec vectors that compose the document (Le and Mikolov, 2014).

Similarity Analysis

Throughout the research, cosine similarities between text data were calculated to investigate the comparable specifications or to extract the corresponding sentences. The most well-known measure of vector similarity would be Euclidean distance, however, it is known that it does not fully reflect the distance between text vectors. For this reason, in this research we used cosine distance to investigate the similarities in the text.

Cosine similarity computes the distance of two vectors based on the inner value of the angle, not the straight distance. By doing so, the excessive frequency of certain words cannot distort the distance between vectors. The cosine similarity between two vectors, *A* and *B*, would be calculated as shown in Equation 5, where *n* indicates the dimension of the vectors, and A_i indicates the value of the i^{th} element of vector *A*.

$$\text{Cosine Similarity} = \cos\theta = \frac{A \cdot B}{\|A\|\|B\|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2}\sqrt{\sum_{i=1}^{n} B_i^2}} \qquad \text{Equation (5)}$$

Named Entity Recognition

Named Entity Recognition (NER) is one concept of text classifications, and it automatically labels each word with informative categories, such as location, name, object, and action (McCallum and Li, 2003). The target categories, called Named Entities, were assigned by researchers, and, in the research, the words were labeled by six categories, i.e., (1) none, (2) object, (3) standard, (4) environment, (5) condition, and (6) reference. The description of each category is provided in Table 1.

Table 1. Word Categories for NER

Category	Description
None	Not an informative element for text analysis
Object	A subject of construction specification standards
Standard	A construction standard stated in the specification
Environment	An environmental factor that affects the construction standard
Condition	A detailed condition of the environmental factor
Reference	A referenced document for the standard

NER is commonly conducted in two different ways, i.e., via the rule-based model and via the machine learning model. The rule-based model performs the recognizing process based on the predetermined rules, such as ‘FHWA → [Organization]’, ‘Ohio → [Region]’, and ‘Asphalt → [Object]’. Because of the definite rules, the accuracy of the model would be considerably high, but the model could not recognize any other entities that were not stated in the rules. To address this limitation, in our research, we conducted NER with the machine learning concept by developing a Recurrent Neural Network (RNN) model. RNN is a concept of Artificial Neural Network (ANN), which is suitable for handling sequential data (Mikolov et al., 2010). The model utilizes the previous classification results recurrently, that is, the input vector of the current step and the output class of the previous step are used as the input to the current step.

EXPERIMENTAL DESIGN AND RESULTS

Research Framework

The overall research framework consists of four steps that are processed by the previously mentioned text mining methodologies: (1) selection of comparable specifications, (2) identification of corresponding sentences, (3) extraction of construction standards, and (4) comparison of construction standards. The research progressed by the third step (i.e., Extraction of Construction Standards) and such interim results were presented in this section.

Step (1) Result: Selection of Comparable Specifications

Remembering that the research objective was automatically reviewing the construction specifications, we first needed a set of reference specifications that were able to be used as the correct answer. We assumed that the standard specifications of the U.S. were well written, and we used them as candidates of the reference data in our research. In addition, considering the prior knowledge that construction specifications commonly consist of a combination of the standard specifications, it seems appropriate to select the most similar specifications for the reference data (i.e., comparable specifications).

The text data of specifications were represented in numeric vectors by TF-IDF embedding, and then we calculated the cosine similarities between the construction specification in Qatar (QAT) and the 43 standard specifications (USA). As a result, the standard specifications from Alabama, Colorado, and Arkansas showed high similarities to QAT, i.e., 0.728, 0.723, and 0.718, respectively. After qualitative investigation by industry practitioners, these three were used as the comparable specifications in the following steps.

Although Word2Vec and Doc2vec commonly are known to dominate the TF-IDF in language modeling and computing efficiency, those models show disadvantages in interpretation because they mix up the vector space while learning the corpus, relationship database of words used in specifications. Therefore, since the results of this ‘Selection of Comparable Specification’ step must be analyzed qualitatively by the practitioners, we embedded TF-IDF method in the specification documents.

Step (2) Result: Identification of Corresponding Sentences

Occasionally, certain standards might have different values depending on the associated category. For instance, the standard value of the thickness of concrete would be different according to whether the category is the ceiling or the floor. Thus, it is crucial to identify the corresponding text (e.g., category, paragraph, and sentence) that describes the same target prior to reviewing the construction standard.

This paper identified corresponding sentences from only two paragraphs that we had concluded correspond to each other for the feasibility testing purpose. The omitted steps, i.e., identifying corresponding categories and paragraphs, will be covered by future research planned by the authors.

In this research, we assumed that the corresponding text would show high similarity, Doc2Vec embedding was conducted for every sentence from four documents (i.e., QAT construction specification and three comparable specifications), and then, we calculated the cosine similarities between each pair of sentences. The results showed insufficient quality, as evidenced by including the correct sentence in the 7th rank among 334 sentences.

Step (3) Result: Extraction of Construction Standards

After the corresponding sentences were identified, the information of object, standard, environment, condition, and reference must be extracted automatically. In this research, we developed an RNN model to recognize the entities from the text data.

Since there is no existing labeled data for NER in the field of construction specification review, the researchers had to label every sentence one by one. For now, only 273 sentences have been labeled and utilized in developing the NER model. We used 70% of the data (191 sentences) to train the model, and the remaining 30% (82 sentences) was used to validate the classification results.

Table 2 is a confusion matrix of the classification results of the NER model. The results in the table indicate that the model failed to classify anything for the categories 'none'. Moreover, the model seemed to be naïve in that it categorized most words into the 'standard' category.

Table 2. Confusion Matrix of NER

		Prediction						
		none	obj	std	env	con	ref	**total**
Actual	none	0	204	517	107	43	1	**872**
	obj	0	174	24	11	9	0	**218**
	std	0	27	425	40	22	0	**514**
	env	0	40	48	32	20	2	**142**
	con	0	17	64	19	20	11	**131**
	ref	0	3	15	0	1	3	**22**
	total	**0**	**465**	**1,093**	**209**	**115**	**17**	**1,899**

To validate the NER model quantitatively, precision and recall were measured as shown in Table 3. As mentioned above, the model categorized most words into the 'standard' category, so that the recall of 'standard' had a high value (0.827). However, the overall results of precision and recall were both inadequate, and the average F1 score was only 0.256. Discussion of these results and our plan for future research are presented in the conclusion section.

Table 3. Precision and Recall (NER)

Class	Precision	Recall	F1 Score
none	0.000 (0/0)	0.000 (0/872)	0.000
obj	0.374 (174/465)	0.798 (174/218)	0.510
std	0.389 (425/1093)	0.827 (425/514)	0.529
env	0.153 (32/209)	0.225 (32/142)	0.182
con	0.174 (20/115)	0.153 (20/131)	0.163
ref	0.176 (3/17)	0.136 (3/22)	0.154

CONCLUSION

Since the research was ongoing, some critical assumptions had to be made, and the interim results were limited with relatively large error rates. In addition, the sizes of the training sentences definitely were insufficient to train the RNN model; the more training sentences we have, better the deep learning functions work. To overcome these problems, in future research, we plan to collect more specifications from Australia and the United Kingdom, expand the training set by labeling additional sentences, and thus enhance the models that were developed.

The results of this research suggested that an automatic reviewing framework of construction specifications was required to cover all of the processes involved, ranging from the collection of data to extracting the target information. In a future study, we will conduct comparison analysis between the standard information from different specifications. In addition, we plan to test the applicability of our approach by applying the research results to several construction sites as case studies.

ACKNOWLEDGMENT

This research was supported by Daewoo Engineering & Construction and the BK21 PLUS research program of the National Research Foundation of Korea, and we appreciated the pre-processing of the data that was performed by researchers in the Construction Innovation Laboratory at Seoul National University.

REFERENCES

Chopra, D., Joshi, N., and Mathur, I. (2016). *Mastering Natural Language Processing with Python*, Packt Publishing Ltd.

Le, Q., and Mikolov, T. (2014). “Distributed Representations of Sentences and Documents.” *31st International Conference on Machine Learning*, Beijing, p. 1–9.

Lee, J., Yi, J.-S., and Son, J. (2016). Unstructured Construction Data Analytics Using R Programming - Focused on Overseas Construction Adjudication Cases: *Journal of the Architectural Institute of Korea*, Vol. 32, No. 5, pp. 37–44, DOI: http://dx.doi.org/10.5659/JAIK_SC.2016.32.5.37.

McCallum, A., and Li, W. (2003). “Early results for named entity recognition with conditional random fields, feature induction and web-enhanced lexicons.” *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003*, Edmonton, Canada, p. 188–191.

Mikolov, T., Corrado, G., Chen, K., and Dean, J. (2013). Efficient Estimation of Word Representations in Vector Space: *Proceedings of the International Conference on Learning Representations (ICLR 2013)*, pp. 1–12, DOI: 10.1162/153244303322533223.

Mikolov, T., Karafiat, M., Burget, L., Cernocky, J.H., and Khudanpur, S. (2010). “Recurrent neural network based language model.” *Eleventh Annual Conference of the International Speech Communication Association*,, p. 1045–1048.

Al Qady, M., and Kandil, A. (2015). Automatic Classification of Project Documents on the Basis of Text Content: *Journal of Computing in Civil Engineering*, Vol. 29, No. 3, pp. 1–11, DOI: 10.1061/(ASCE)CP.1943-5487.0000338.

Zhang, J., and El-Gohary, N.M. (2016). Semantic NLP-Based Information Extraction from Construction Regulatory Documents for Automated Compliance Checking: *Journal of Computing in Civil Engineering*, Vol. 30, No. 2, pp. 1–14, DOI: 10.1061/(ASCE)CP.1943-5487.0000346.