

Received December 9, 2020, accepted January 8, 2021, date of publication January 13, 2021, date of current version January 26, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3051237

Learning-Free Unsupervised Extractive Summarization Model

MYEONGJUN JANG¹ AND PILSUNG KANG²

¹Department of Computer Science, University of Oxford, Oxford OX1 3QD, U.K.

²School of Industrial Management Engineering, Korea University, Seoul 02841, South Korea

Corresponding author: Pilsung Kang (pilsung_kang@korea.ac.kr)

This work was supported in part by the National Research Foundation of Korea (NRF) Grants Funded by the Korean Government (MSIT) under Grant NRF-2019R1F1A1060338 and in part by the Korea Institute for Advancement of Technology (KIAT) Grant Funded by the Korean Government (MOTIE) (The Competency Development Program for Industry Specialist) under Grant P0008691.

ABSTRACT Text summarization is an information condensation technique that abbreviates a source document to a few representative sentences with the intention to create a coherent summary containing relevant information of source corpora. This promising subject has been rapidly developed since the advent of deep learning. However, summarization models based on deep neural network have several critical shortcomings. First, a large amount of labeled training data is necessary. This problem is standard for low-resource languages in which publicly available labeled data do not exist. In addition, a significant amount of computational ability is required to train neural models with enormous network parameters. In this study, we propose a model called Learning Free Integer Programming Summarizer (LFIP-SUM), which is an unsupervised extractive summarization model. The advantage of our approach is that parameter training is unnecessary because the model does not require any labeled training data. To achieve this, we formulate an integer programming problem based on pre-trained sentence embedding vectors. We also use principal component analysis to automatically determine the number of sentences to be extracted and to evaluate the importance of each sentence. Experimental results demonstrate that the proposed model exhibits generally acceptable performance compared with deep learning summarization models although it does not learn any parameters during the model construction process.

INDEX TERMS Text summarization, natural language processing, sentence representation vector, integer linear programming.

I. INTRODUCTION

Text summarization is the task of condensing a source document into a short and concise set of sentences while preserving the predominant information of the original text. There are two approaches when it comes to text summarization: *extractive* and *abstractive*. Extractive summarization composes summaries by selecting representative sentences or phrases from the source text, whereas abstractive summarization generates new sentences or phrases that concisely convey the staple topic of the source document.


Recent developments in deep learning have brought rapid progress in text summarization. The release of publicly available large labeled summarization corpora such as the CNN/Daily Mail dataset [1]–[3] has enabled the training of deep learning models with a large number of parameters. Among them, recurrent neural network (RNN) and

convolution neural network (CNN) have been widely used for text summarization.

In extractive methods, RNN is employed to evaluate sentence importance while selecting representative sentences simultaneously [4]. It is also used in the approach where sentences are considered latent variables [5]. Kedzie *et al.* [6] leveraged the average of word vectors, RNN, and CNN to represent sentences to distributed vectors. The performance of each sentence encoder was evaluated for both RNN and sequence-to-sequence (seq2seq) sentence extractors.

When it comes to abstractive models, Rush *et al.* [7] used the seq2seq structure [8] and the attention mechanism [9] for effective summary generation through the context vector. See *et al.* [3] proposed a method that combines seq2seq attention structure, pointer networks [10], and coverage modeling [11]. Liu *et al.* [12] applied generative adversarial networks [13] to the method of See *et al.* [3] for enhanced performance.

Although deep neural networks (DNNs) have made substantial progress in text summarization, they have a

The associate editor coordinating the review of this manuscript and approving it for publication was Arif Ur Rahman .

critical shortcoming: the necessity of a tremendous amount of labeled training data. As publicly available summarization corpora are mainly written in English, a new dataset is required to apply state-of-the-art deep learning models in other low-resource languages, which is a time- and resource-consuming task. In addition, sufficient computation resources are a prerequisite for training the DNN-based summarization model. Moreover, to achieve high performance, deep learning models should improve their generalization ability through large training datasets. This implies that although models can effectively react to general input, they are unable to process a novel pattern that does not appear in the training dataset. By contrast, humans can flexibly react to unusual patterns. Humans are capable of summarizing a document without learning through labeled summarization datasets provided they have a basic understanding of the language.

In this paper, we propose a novel learning-free integer programming summarizer named LFIP-SUM. It is a strong advantage that our approach is a training-free model. Therefore, neither labeled training data nor a large number of computational resources are required. Our practical approach of leveraging significantly fewer resources than DNN-based methods can be very efficient when applied in the real industry, where available resources are limited.

To achieve this purpose, it is vital to make the model understand natural language and select a few representative sentences from a given document. Therefore, we leveraged publicly available DNN-based pre-trained sentence embedding vectors, which act as the aforementioned basic language understanding of humans. Employing pre-trained sentence vectors is of benefit to reducing resource consumption because they could be trained by either unsupervised methods [14]–[17] or supervised classification tasks [18], [19], which require significantly fewer time and fortune to obtain labeled data than summarization tasks. Next, in order to select representative sentences, we used integer linear programming (ILP), which is a widely used method in industrial engineering and has been applied to extractive summarization [20]–[24]. Additionally, we used principal component analysis (PCA) to automatically determine the most appropriate number of summary sentences. Through an experiment, it was verified that the proposed model exhibited comparable performance to that of the supervised state-of-the-art model. The main contributions of this paper can be summarized as follows:

- In this paper, we propose a document summarization framework that does not require any model training. Hence, our approach requires neither human-labeled training data nor a large number of computational resources.
- Our approach can be applied to any language source, including low-resource language, because it is free from the necessity of labeled data.
- We propose an approach that dynamically determines the number of summary sentences in terms of intrinsic information preservation.

The remainder of this paper is organized as follows. In section 2, we briefly describe previous *extractive* summarization researches including ILP-based and DNN-based methods. In section 3, we describe the structure of LFIP-SUM model and the experimental results are illustrated in section 4. Finally, in section 5, we conclude the present work with some discussion about future research directions.

II. RELATED WORKS

McDonald [20] proposed the first ILP method for extractive summarization. As shown below, it generates summaries by maximizing the relevance (i.e., importance) of the selected sentences and minimizing their redundancy (i.e., similarity):

$$\max_{x,y} \sum_{i=1}^n \text{imp}(s_i) \cdot x_i - \sum_{i=1}^n \sum_{j=i+1}^n \text{sim}(s_i, s_j) \cdot y_{i,j}, \quad (1)$$

$$\text{subject to } \sum_{i=1}^n l_i \cdot x_i \leq L_{\max}, \quad (2)$$

and for $i = 1, \dots, n$ and $j = i + 1, \dots, n$

$$\begin{aligned} y_{i,j} - x_i &\leq 0 \\ y_{i,j} - x_j &\leq 0 \\ y_i + x_j - y_{i,j} &\leq 1, \end{aligned} \quad (3)$$

where $\text{imp}(s_i)$ is the importance score of sentence s_i , n is the number of sentences in the source document, l_i is the length of s_i , $\text{sim}(s_i, s_j)$ is the similarity between s_i and s_j , and L_{\max} is the maximum length of summary sentences. x_i is a binary variable indicating whether the sentence s_i is selected in the summary. $y_{i,j}$ denotes a binary variable indicating whether both s_i and s_j are included in the summary. McDonald [20] represented each sentence as a bag-of-words vector with TF-IDF values. The importance scores are computed by using the positional information of the sentences and the similarity between each sentence vector and the document vector. The cosine similarity is used to compute the similarity between sentence vectors.

Berg-Kirkpatrick *et al.* [21] constructed an ILP summarization model based on the notion of *concept*, which is actually a set of bi-grams. The distinctive characteristic of this model is that it extracts and compresses sentences simultaneously. The model not only selects bi-grams with high importance but also chooses whether to cut (delete) individual subtrees from each sentence's parsing tree. The objective function of this model is the following:

$$\max_{b,c} \sum_{i=1}^{|B|} w_i \cdot b_i + \sum_{i=1}^{|C|} u_i \cdot c_i, \quad (4)$$

where b_i and c_i are binary variables that indicate the selection of the i^{th} bi-gram as a summary and its deletion from the parsing tree. w_i and u_i indicate the weights of bi-grams and possible subtree cuts, respectively. Additionally, the model has a constraint of maximum allowed summary length, which

is determined by the user. The weights are estimated by soft-margin support vector machine optimization with bi-gram recall loss function. Therefore, the model is trained in a supervised manner, which requires gold-standard summaries.

Galanis *et al.* [23] also presented a supervised extractive summarization model that extracts sentences and concepts by maximizing sentence importance and diversity (i.e., minimizing redundancy). To represent sentences in a structured form, they leveraged various features, such as sentence position, named entities, word overlap, content word frequency, and document frequency. As in previous studies, the model has a constraint of user-defined maximum summary length. Furthermore, support vector regression (SVR) was used to estimate sentence importance. The SVR was trained to predict the average ROUGE score [25] between source sentences and their gold-standard summaries.

More recently, Boudin *et al.* [24] proposed a purely concept-based extractive summarization model. The objective function of this model is the following:

$$\max_i \sum_i w_i \cdot c_i + \mu \sum_k f_k \cdot t_k, \quad (5)$$

where w_i is the weight of a concept, f_k is the frequency of the non-stop word k in the document set, c_i is a binary variable indicating whether the concept i is selected, and t_k is a binary variable indicating the presence of the non-stop word k in the summary. This variable was introduced because the frequency of a non-stop word is a good predictor of a word appearance in a human-generated summary. To obtain the concept weight, heuristic counting (such as the document frequency of each concept) was used, or the supervised model was trained. The model also has a user-determined maximum summary length. Boudin *et al.* [24] differs from previous studies in that it applied sentence pruning. Sentences with fewer than 10 words were removed to improve computational efficiency.

Liu *et al.* [26] proposed a simple weighting-based unsupervised extractive summarization method. For the i^{th} sentence in a document (S_i), they calculated sentence scores by leveraging term frequency, similarity, positional information, and sentence length. In particular, term frequency, position score, and sentence length score are calculated as follows:

$$\begin{aligned} TF_i &= \sum_{w \in \text{sen}_i} tf(w), \\ Position_i &= \frac{n - p_i + 1}{n}, \\ Length_i &= \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x_i - \mu)^2}{2\sigma^2}}, \end{aligned} \quad (6)$$

where $tf(w)$ denotes the term frequency of the word w , n represents the total number of sentences, p_i refers to the position of S_i , x_i is the length of S_i , and μ and σ denote the mean and standard deviation of the total sentence length, respectively. Unlike the three score metrics mentioned above, which can be calculated by leveraging the information contained within a document, the similarity score is gained using external

information: the title of a news article. For a given sentence vector \vec{s} and title vector \vec{t} , similarity score is defined as the cosine similarity of two vectors:

$$Similarity_i = \frac{\vec{s} \cdot \vec{t}}{|\vec{s}| |\vec{t}|}. \quad (7)$$

The vectors of each sentence and article title are built based on the term frequency weighting schema. Therefore, the final sentence score is calculated as follows:

$$Score_i = \lambda_1 TF_i + \lambda_2 Position_i + \lambda_3 Similarity_i + \lambda_4 Length_i, \quad (8)$$

where λ_1 , λ_2 , λ_3 , and λ_4 are user-defined hyper-parameters. Liu *et al.* [26] generated a summary by selecting sentences with the highest score until the total summary did not exceed the user-defined summary length.

The abovementioned approaches have certain common disadvantages. First of all, they leverage only the low-level features of documents, such as n-gram, stop-words, and term-frequency, to attain a machine-generated summary. These representations lack semantic information compared to the distributed representation of words and sentences, making it difficult to analyze more elaborate and refined documents. In addition, as the number of documents increases, the vector dimension of the n-gram or term-frequency representation soars exponentially, which results in the high computational cost and the curse of dimensionality.

The recent development of DNNs has effectively overcome the abovementioned limitations. Nallapati *et al.* [27] proposed an RNN-based extractive summarization model called *Summarunner*. Instead of using low-level features, they used two RNN layers for encoding documents: a word-level encoder, and a sentence-level encoder. They then trained the binary sequential labeling model in which each sentence should be included in the summary.

Zhang *et al.* [5] proposed a more sophisticated extractive summarization model called *NeuSum*, which jointly selects and scores summary candidate sentences. On the top of a document encoder, which has a hierarchical structure similar to that of *Summarunner*, Zhang *et al.* [5] added a GRU-based sequential sentence extractor that predicts a summary inclusion score. In particular, the score of S_i , the i^{th} sentence, at time t is calculated as follows:

$$\begin{aligned} h_t &= \text{GRU}(s_{t-1}, h_{t-1}), \\ \delta(S_i) &= \vec{W}_s \tanh(\vec{W}_q h_t + \vec{W}_d s_i), \end{aligned} \quad (9)$$

where \vec{W}_s , \vec{W}_q , and \vec{W}_d are trainable parameters; s_{t-1} is a sentence vector of a previously selected sentence; and h_t is the hidden state of the sentence extractor at time t . To train the model, Zhang *et al.* [5] minimized the Kullback-Leibler (KL) divergence of model prediction and labeled training data distribution. The model prediction distribution P is calculated as follows:

$$P(\hat{S}_t = S_i) = \frac{\exp(\delta(S_i))}{\sum_{k=1}^L \exp(\delta(S_k))}. \quad (10)$$

The labeled data distribution is derived from the relative ROUGE F1 gain at time step t provided by previously selected sentence S_{t-1} . In particular, the ROUGE F1 gain is calculated as follows:

$$\begin{aligned} g(S_i) &= r(S_{t-1} \cap S_i) - r(S_{t-1}), \\ \tilde{g}(S_i) &= \frac{g(S_i) - \min(g(S))}{\max(g(S)) - \min(g(S))}. \end{aligned} \quad (11)$$

Finally, the labeled distribution Q is derived from a temperature-scaled softmax distribution:

$$Q(S_i) = \frac{\exp(\tau \tilde{g}(S_i))}{\sum_{k=1}^L \exp(\tau \tilde{g}(S_k))}. \quad (12)$$

Zhang *et al.* [5] proposed a model that considers sentences in a document as latent variables. The latent variable $z_i \in \{0, 1\}$ denotes whether the i^{th} sentence (S_i) should be selected as a summary. They first trained two models: (1) an extractive summarization model with an RNN-based hierarchical structure similar to previous works and (2) a sentence compression model. The role of the sentence compression model is to map a selected extractive sentence S_i to a human-generated gold summary H_j . To achieve this, they generated training instances $\langle S_i, H_j \rangle$ that maximize the ROUGE score between them. They then trained a compression model with a standard attention-based sequence-to-sequence architecture [28]. Next, they sampled latent variables z_i from the extractive summarization model and obtained $C = (C_1, \dots, C_{|C|})$, a set of sentences whose latent variable are equal to 1. Thereafter, they estimated the likelihood of summary sentence H_i being the compression of C_j by leveraging the compression model. The likelihood is calculated as follows:

$$s_{ji} = \exp\left(\frac{1}{|H_i|} \log p_{s2s}(H_i|C_j)\right), \quad (13)$$

where $\log p_{s2s}(H_i|C_j)$ is the probability of a sentence H_i being the compression of C_j . They defined two metrics $R_p(C, H)$ and $R_r(C, H)$, based on this score. The former measures the extent to which the summary sentence H can be inferred from C , whereas the latter denotes the extent to which C is compressed to H :

$$\begin{aligned} R_p(C, H) &= \frac{1}{|C|} \sum_{j=1}^{|C|} \max_i s_{ji}, \\ R_r(C, H) &= \frac{1}{|H|} \sum_{i=1}^{|H|} \max_j s_{ji}. \end{aligned} \quad (14)$$

The final score $R(C, H)$ is a weighted sum of two metrics, and the model is trained to minimize the negative expectation of $R(C, H)$:

$$L = -E[\alpha R_p(C, H) + (1 - \alpha) R_r(C, H)]. \quad (15)$$

While the aforementioned DNN-based models demonstrate a promising performance, there is a disadvantage in these approaches: the necessity of human-labeled training data. Contrary to in the case of English, for which plenty

of summarization datasets exist, this disadvantage could be disastrous for other low-resource languages because a significant effort would be required to build a human-labeled dataset. In this paper, we propose a learning-free extractive summarization model to overcome this disadvantage.

III. PROPOSED MODEL

In this section, we present the LFIP-SUM model. It consists of a two-step procedure: document representation and representative sentence selection. To represent a document as a continuous vector, we use distributed vectors pre-trained by deep learning-based sentence embedding models. Next, ILP and PCA are used to evaluate the sentence importance score and select the representative sentences for the summary. The implementation of our approach is available at https://github.com/MJ-Jang/LFIP_SUM. We further improved model performance by an ensemble of different pre-trained sentence representations. The overall structure of the model is shown in Figure 1.

A. DEEP REPRESENTATION OF A DOCUMENT

In previous studies on ILP-based extractive summarization, sentence representation generally relies on simple word counting, which can hardly capture the intrinsic meaning of sentences. In this study, we used deep representation vectors to capture the latent meaning of each sentence. We assume that a document consists of n sentences. As the sentences have sequential information, we can represent a document as a sequence of sentences as follows:

$$D = [s_1, s_2, \dots, s_n], \quad (16)$$

where D denotes the document and s_k refers to its k -th sentence. Let a sentence be represented as a column vector. Thus, D_{basic} , the basic representation of D , becomes the following matrix:

$$\mathbf{D}_{\text{basic}} = [\mathbf{sv}_1, \mathbf{sv}_2, \dots, \mathbf{sv}_n], \quad (17)$$

where \mathbf{sv}_k denotes the pre-trained sentence vector of s_k . $\mathbf{D}_{\text{basic}}$ is a $d \times n$ matrix, where d is the embedding dimension. It is possible to use any sort of sentence embedding method to generate \mathbf{sv}_k .

Although the matrix $\mathbf{D}_{\text{basic}}$ contains the intrinsic meaning of each sentence, it lacks positional information, which plays a critical role in natural language tasks. Therefore, we used positional encoding to effectively reflect sequential information. We adopted the positional encoding method used in the transformer [29], which employs cosine and sine functions. In particular, the positional encoding matrix PE is calculated as follows:

$$\begin{aligned} PE_{(pos, 2i)} &= \sin(pos/10000^{2i/d}), \\ PE_{(pos, 2i+1)} &= \cos(pos/10000^{2i/d}), \end{aligned} \quad (18)$$

where pos is the position, and i is the dimension. Then, the final input embedding matrix D_{emb} is calculated as the

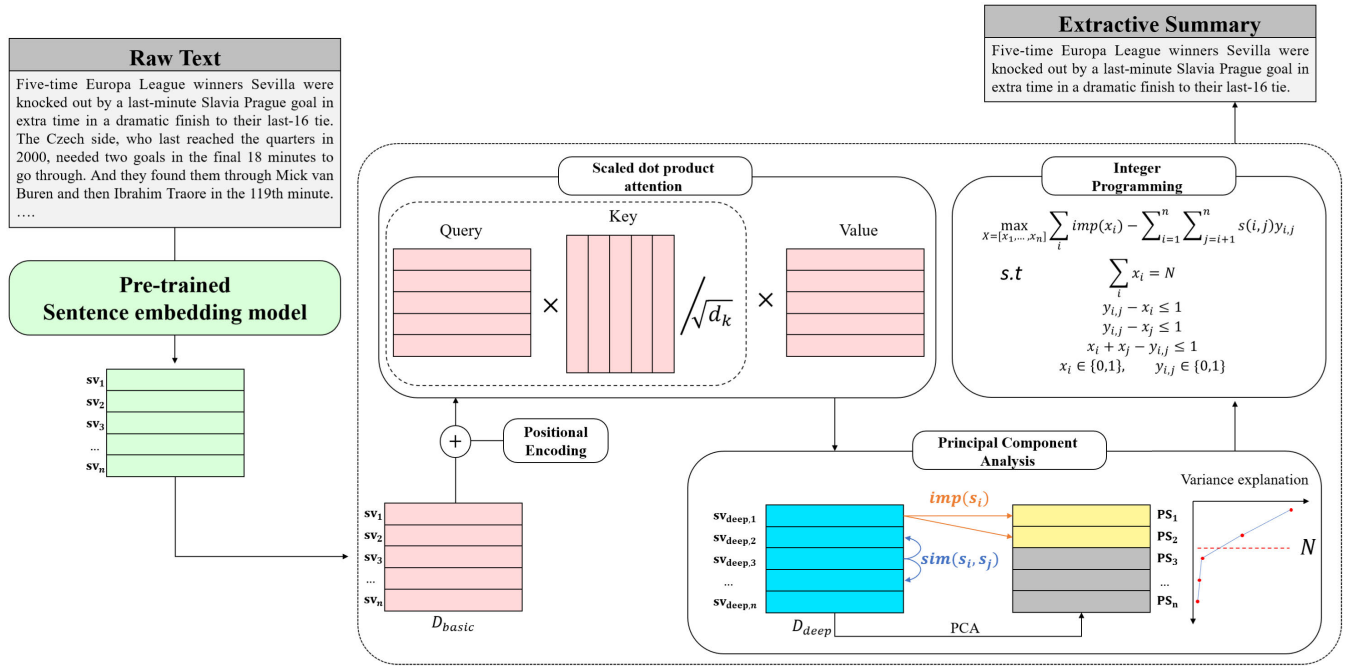


FIGURE 1. Structure of LFIP-SUM model. First, the document is represented as a stack of pre-trained sentence vectors. Thus, deep representation of the document is generated by leveraging positional encoding and self-attention mechanism. Next, the number of summary sentences, which preserve at least $\beta\%$ of information, is automatically determined through PCA. Subsequently, sentence importance and sentence similarity are calculated by leveraging deep representation of document and extracted PCs. Finally, summary sentences are extracted through ILP with help of computed sentence importance and similarity.

addition of D_{basic} and PE .

$$D_{\text{emb}} = D_{\text{basic}} + PE. \quad (19)$$

Subsequently, to obtain a deeper representation of the document, we used scaled dot product attention [29], whereby attention weights can be calculated without training parameters.

The matrix \mathbf{Q} is regarded as a set of queries. Likewise, the matrices \mathbf{K} and \mathbf{V} are the sets of keys and values, respectively. Then, the result of the scaled dot product attention is as follows:

$$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_k}}\right)\mathbf{V}, \quad (20)$$

where d_k is the dimension of queries and keys. As we use the self-attention method, the matrices \mathbf{Q} , \mathbf{K} , and \mathbf{V} are \mathbf{D}_{emb} . Therefore a deep representation \mathbf{D}_{deep} of the document is calculated as

$$\mathbf{D}_{\text{deep}} = \text{Attention}(\mathbf{D}_{\text{emb}}, \mathbf{D}_{\text{emb}}, \mathbf{D}_{\text{emb}}), \quad (21)$$

where \mathbf{D}_{deep} has dimension $d \times n$, the same dimension as that of \mathbf{D}_{emb} . Therefore, the matrix \mathbf{D}_{deep} can be considered the sequence of the deep representation vectors of the sentences constituting the document.

$$\mathbf{D}_{\text{deep}} = [\mathbf{sv}_{\text{deep},1}, \mathbf{sv}_{\text{deep},2}, \dots, \mathbf{sv}_{\text{deep},n}]. \quad (22)$$

In this study, we used four different pre-trained sentence embedding models: SIF [15], InferSent [18], Universal sentence encoder (USE) [19], and BERT [30]. Unlike SIF,

InferSent, and USE, which generate a sentence vector, BERT generates contextualized represents of sentence as a $T \times d_h$ matrix, where T is the number of tokens and d_h is the dimension of the hidden states. As the representation of a sentence should be a vector to be applied in our study, we consider the average of the BERT representations to obtain the dimension d_h .

B. PRINCIPAL COMPONENT ANALYSIS

A common characteristic of previous studies on extractive summarization using ILP is that they set the maximum length of summary sentences as a hyper-parameter. However, in practice, the number of summary sentences varies. Some documents may provide a summary that reflects all information using a single sentence, whereas others may require more sentences. Accordingly, the proposed LFIP-SUM automatically determines the number of appropriate summary sentences for each document by PCA, and thus the importance of each sentence can be quantitatively evaluated.

PCA is a method for reducing high-dimensional data to lower dimensions [31]. The principal components (PCs) extracted by PCA are composed of a linear combination of the original variables. Hence, PCA has been widely used for dimensionality reduction because it is possible to explain the entire dataset through a few PCs. The purpose of PCA is to maximize the variance of $\mathbf{y} = \mathbf{X}\mathbf{w}$, which is the projection of the original data \mathbf{X} , where \mathbf{X} is an $n \times p$ matrix and \mathbf{w} is a vector of size $p \times 1$. n is the number of observations and p is the number of variables. For a centered dataset \mathbf{X} , PCA is

performed by the following optimization:

$$\begin{aligned} \text{Max } \text{Var}(\mathbf{Y}) &= \mathbf{w}^T \mathbf{\Sigma} \mathbf{w}, \\ \text{s.t. } \|\mathbf{w}\| &= 1, \end{aligned} \quad (23)$$

where $\mathbf{\Sigma}$ is the covariance matrix of \mathbf{X} . Solving the above equation for \mathbf{w} yields the eigenvectors \mathbf{e} of the covariance matrix. Therefore, when the i -th largest eigenvalue λ_i and its corresponding eigenvector \mathbf{e}_i are given, the i -th PC is calculated as follows:

$$\mathbf{y}_i = \mathbf{e}_{i1}\mathbf{X}_1 + \mathbf{e}_{i2}\mathbf{X}_2 + \dots + \mathbf{e}_{ip}\mathbf{X}_p. \quad (24)$$

Furthermore, by the following equation, the variance of \mathbf{y}_i is λ_i .

$$\text{Var}(\mathbf{y}_i) = \mathbf{e}_i^T \mathbf{\Sigma} \mathbf{e}_i = \lambda_i. \quad (25)$$

As the total population variance is $\sum_{i=1}^p \lambda_i$, the variance preservation ratio of \mathbf{y}_i is

$$\alpha_i = \frac{\lambda_i}{\sum_{i=1}^p \lambda_i}, \quad (26)$$

where α_i is the variance preservation ratio of \mathbf{Y}_i . This implies that a PC \mathbf{y}_i preserves $(100 \times \alpha_i)\%$ of the total variance.

The purpose of text summarization is to condense a document composed of several sentences so that a concise version may be obtained. As \mathbf{D}_{deep} has sentence vectors as column vectors, we could consider each sentence vector as a variable and the dimension of the vector, d , as the number of observations. Therefore, we can use PCA to reduce the number of sentences in a document by extracting the PCs. Each PC, which we refer to as a principal sentence (PS) in this paper, could be considered a vector of condensed sentences that contains as much information of the original document as possible. Subsequently, we define a user-specific hyper-parameter β —the variance preservation ratio—to automatically decide the appropriate number of summary sentences N . Therefore, it could be considered that at least $\beta\%$ of the total information is preserved in N selected sentences.

We then calculated the sentence importance score based on the correlation between the sentence vectors of \mathbf{D}_{deep} and the PSs. As a PS is a high-level vector that efficiently condenses the intrinsic information of the document, greater similarity of a sentence vector to a PS implies higher importance of the sentence. Therefore, we defined the sentence importance score as follows:

$$\text{imp}(s_i) = \sum_{k \neq i} \cos(\mathbf{sv}_{\text{deep},i}, \mathbf{PS}_k), \quad (27)$$

where $\text{imp}(s_i)$ is the sentence importance score of sentence s_i and \cos denotes cosine similarity.

C. INTEGER LINEAR PROGRAMMING

1) FORMULATION

We formulate the optimization problem just as in Eq. (1)-(3), because the minimum extraction unit in this study is a sentence, not a concept. The requirements for the formulation

are sentence importance scores and similarity scores between sentences. We define the similarity between the sentences s_i and s_j as the cosine similarity of their deep sentence representation vectors as follows:

$$\text{sim}(s_i, s_j) = \cos(\mathbf{sv}_{\text{deep},i}, \mathbf{sv}_{\text{deep},j}). \quad (28)$$

Then, unlike in McDonald [20], the appropriate number of summary sentences (the length of summary sentences) for each document is automatically determined through PCA. Therefore, we modify Eq. (2) as follows:

$$\sum_{i=1}^n x_i = N, \quad (29)$$

where N is the appropriate number of summary sentences.

2) SENTENCE PRUNING

A major shortcoming of the above ILP formulation is that its time complexity is $O(n^2)$, where n is the number of sentences in the document. To reduce the time complexity of LFIP-SUM, sentence pruning [24] is performed to remove unimportant sentences. According to the above equations, the sentences to be extracted are those with high sentence importance scores and low redundancy scores. Therefore, we define the pruning score of a sentence s_i as follows:

$$\text{Pr} - \text{score}(s_i) = \frac{\text{imp}(s_i)}{\frac{1}{n-1} \sum_{j \neq i} \text{sim}(s_i, s_j)}. \quad (30)$$

The hyper-parameter for sentence pruning is the maximum sentence number l . For documents with length less than or equal to l , sentences are not pruned, whereas for documents with more than l sentences, ILP is performed on the top l sentences based on their $\text{Pr} - \text{score}$.

D. ENSEMBLE

Ensemble modeling is a method of merging several models trained with the same objective; it is widely used in machine learning [32]–[34]. The key components to maximize the performance of an ensemble model are ensuring a good performance of individual models above a certain level and assuring low correlations between the constituent models. In this study, we construct four models with different sentence embeddings, which ensures low correlations between the models. Accordingly, we construct the final model as an ensemble of the four summarization models. In particular, we calculated the sentence importance and sentence similarity using the average values over four models and leveraged them for extracting the candidate sentences.

IV. EXPERIMENTAL RESULTS

We conducted experiments on three different publicly available summarization datasets: a non-anonymized version of the CNN/Daily Mail dataset¹ [3], the Wikihow dataset² [35],

¹https://huggingface.co/datasets/cnn_dailymail

²<https://github.com/mahnazkoupae/WikiHow-Dataset>

and the Cornell Newsroom dataset³ [36]. The addresses of the web pages from which we collected the datasets have been provided in footnotes.

The hyper-parameters of the LFIP-SUM model are the minimum variance preservation ratio β and the sentence pruning parameter l . We set β to 0.8 for the InferSent embeddings and 0.9 for the SIF and USE embeddings because most documents are summarized to a single sentence when β is set to 0.8 for the SIF and USE models. l is set to 35 for all three models. As a result, sentence pruning is applied to only 7% of total testing data. We determined the hyper-parameters by employing a validation dataset. In particular, we select β —a value that maximizes the performance on the validation dataset. Regarding l , we determined the value that does not exceed at most 2s for extracting summary sentences using our computing resource (Intel core i7 with 32GB main memory). No training data is used because parameter training is unnecessary for the LFIP-SUM model.

For evaluation, we leveraged the ROUGE score [25], a widely used quantitative metric for the quality of text summarization. The ROUGE score is calculated through the overlapping words between the system and reference summaries. Assume that $R = (r_1, r_2, \dots, r_i)$ is a reference summary and $S = (s_1, s_2, \dots, s_j)$ is a system summary, where r_i and s_j are i^{th} and j^{th} words of the reference and system summaries, respectively. Thus, the precision, recall, and *F1-Score* of ROUGE-N are calculated as follows:

$$\begin{aligned} Precision &= \frac{n(R \cap S)}{n(S)}, \\ Recall &= \frac{n(R \cap S)}{n(R)}, \\ F1 - Score &= \frac{2 \times Precision \times Recall}{Precision + Recall}, \end{aligned} \quad (31)$$

where N denotes the N -gram unit and $n(X)$ indicates the size of set X . For instance, if $N = 1$, the metric calculates the overlap of uni-gram words. In particular, ROUGE-L refers to the metric that leverages the overlap of the longest matching sequence of words. In the experiments, we used the *F1-Score* of ROUGE-1, ROUGE-2, and ROUGE-L for the evaluation.

A. CNN/DAILY MAIL DATASET EXPERIMENT

Table 1 shows the performance comparison between LFIP-SUM and other deep learning-based summarization models on the same dataset. NeuSum [4] is the extractive model exhibiting the highest performance on the non-anonymized CNN/Daily mail dataset. All models except LFIP-SUM are trained in a supervised manner, which requires large training datasets.

The experimental results demonstrate that the LFIP-SUM model generally performed similarly or slightly better than the seq2seq attention structure. However, this does not imply that LFIP-SUM is superior to seq2seq attention models because the former is an extractive summarization model,

TABLE 1. Results obtained on CNN/Daily mail dataset. R-1, R-2, and R-L denote F1-Score of Rouge-1, Rouge-2, and Rouge-L, respectively. 'Abs' refers to abstractive method, whereas 'Ext' denotes extractive methods.

Model	R-1	R-2	R-L
Seq2seq+atten (Abs) (150k voca) [3]	30.49	11.17	28.08
Seq2seq+atten (Abs) (50k voca) [3]	31.33	11.81	28.83
Pointer-generator (Abs) [3]	36.44	15.66	33.42
Pointer-generator +coverage (Abs) [3]	39.53	17.28	36.38
Latent (Ext) [5]	41.05	18.77	37.54
NeuSUM (Ext) [4]	41.59	19.01	37.98
LFIP-SUM-Single best (Ext)	31.10	10.14	20.33
LFIP-SUM-Ensemble (Ext)	36.45	14.29	24.56

TABLE 2. Results obtained on Wikihow dataset. R-1, R-2, and R-L denote F1 of Rouge-1, Rouge-2, and Rouge-L, respectively. 'Abs' refers to abstractive method, whereas 'Ext' denotes extractive methods.

Model	R-1	R-2	R-L
TextRank (Ext) [38], [39]	27.53	7.40	20.00
Seq2seq+atten (Abs) [2], [40]	22.04	6.27	20.87
Pointer-generator (Abs) [3]	27.30	9.1	25.65
Pointer-generator +coverage (Abs) [3]	28.53	9.23	26.54
LFIP-SUM-Single best (Ext)	22.77	4.79	17.10
LFIP-SUM-Ensemble (Ext)	24.28	5.32	18.69

whereas the latter are abstractive summarization models, and the ROUGE measure may be biased toward extractive summarization results. We note that the ROUGE *F1-Score* of the proposed LFIP-SUM model was approximately 76% of that of the NeuSUM model, which is the state-of-the-art performance among extractive summarization models. As the NeuSUM model uses large training datasets with gold-standard summaries, the performance of LFIP-SUM, which is an unsupervised method, is quite meaningful. Considering that the performance of early unsupervised neural translation models was 53% of that of the state-of-the-art supervised model at that time [37], the proposed LFIP-SUM model achieved a significant improvement. We also compared the performance between the single-best LFIP-SUM model and the ensemble LFIP-SUM model and confirmed that the ensemble method significantly contributes to the performance improvement.

B. WIKIHOW DATASET EXPERIMENT

Subsequently, we conducted an experiment on the Wikihow dataset [35]. As this is a fairly recent dataset that has not been extensively tested, we used the results in [35] for comparison. The present results are summarized in Table 2. *TextRank* is an extractive summarization model that uses graph-based ranking. The results demonstrate a similar tendency as in the CNN/Daily Mail dataset. LFIP-SUM exhibited similar performance to that of the seq2seq attention model. As in the CNN/Daily Mail dataset, the ROUGE f-score of LFIP-SUM was 71% of that of *Pointer-generator* with coverage

³<https://github.com/lil-lab/newsroom>

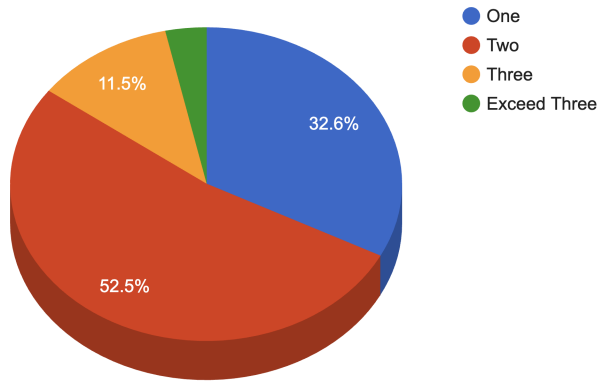


FIGURE 2. Ratio of length of selected sentences. For instance, it means that 52.5% of documents included in CNN-Daily mail, Wikihow, and Cornell newsroom datasets are summarized in 2 extractive sentences.

TABLE 3. Results obtained on Cornell Newsroom dataset. R-1, R-2, and R-L denote F1 of Rouge-1, Rouge-2, and Rouge-L, respectively. 'Abs' refers to abstractive method, whereas 'Ext' denotes extractive methods.

Model	R-1	R-2	R-L
Seq2seq+atten (Abs) [7]	5.91	0.43	5.36
TextRank (Ext) [39]	22.76	9.80	18.97
Pointer-generator +coverage (Abs) [3]	26.04	13.24	22.45
LFIP-SUM-Single best (Ext)	26.30	14.74	19.91
LFIP-SUM-Ensemble (Ext)	31.87	19.39	25.07

mechanism. The ensemble approach improved the performance, but the improvement was not dramatic as that of the CNN/Daily Mail dataset.

C. CORNELL NEWSROOM DATASET EXPERIMENT

Finally, we conducted an experiment on the Cornell newsroom dataset [36]. As this is also quite recent, we used the results at the webpage,⁴ where the dataset was introduced. The present results are provided in Table 3. The results reveal that LFIP-SUM outperformed the other supervised deep learning models. Also, the ensemble approach yielded the significant performance increase. Moreover, we ascertained that the LFIP-SUM model exhibited consistently good performance on all three datasets.

Two summary samples are shown in Table 4. Sentences selected to form a summary are marked in bold. In example 1, which exhibited a high ROUGE score, sentences with exactly the same content as that of the gold-standard summary are extracted. By contrast, example 2 in Table 4 exhibited a low ROUGE score. However, the extracted summary has a similar intrinsic meaning to that of the gold summary. It can be easily verified that both summaries have nearly the same meaning.

D. RESULT ANALYSIS

After the experiment, we analyzed how many sentences per each document were extracted, and Figure 2 shows this percentage. 96.6% of the total number of documents are summarized in less than three sentences. In addition, approximately

TABLE 4. Sample results of LFIP-SUM on English datasets. Extracted summary sentences are formatted in bold.

Example. 1

Document and extracted summary

This is the surreal moment a dog takes the reins and literally drags his owner out for a walk. **Dashcam footage, shared by dailymotion.com user Vidsking, shows a giant St. Bernard running across a road somewhere in the Czech Republic with a child trailing behind.** It appears to be a rather uncomfortable excursion, with the young boy going along the ground on his belly with his legs stretched out behind. As the canine scampers along, his passenger holds tightly to a leash. Two men watching the bizarre scene from their car are heard chuckling in the background. They're forced to slow down to avoid hitting the animal and human train. Once the dog reaches the other side of the road, it makes its way over a muddy verge. At that point the boy gets up and stumbles forwards. Another pedestrian is seen on the roadside but he doesn't appear to be overly fazed by the scene. According to the video time stamp, the incident took place on March 3 just past 5pm. 'A dog walks a child,' the cameraman casually titled the unusual piece of footage. Caught on camera: Dashcam footage shared by dailymotion.com user Vidsking, shows a giant St. Bernard dog running across a road somewhere in the Czech Republic with a child trailing behind. **Bumpy ride: It appears to be a rather uncomfortable excursion, with the young boy being pulled along the ground on his belly with his legs stretched out behind.** Ready for walkies? As the canine scampers along, his passenger holds tightly to a leash"

Gold-standard summary

Dashcam footage shared by dailymotion.com user Vidsking, shows a St. Bernard dog running across a road somewhere with a child trailing behind. It appears to be a rather uncomfortable excursion, with the young boy being pulled along the ground on his belly

Example. 2

Document and extracted summary

England ace Joe Hart labelled fellow goalkeeper Gianluigi Buffon a 'legend of the game' after seeing the Italian veteran claim his 147th cap. Hart, who passed an impressive milestone of his own by representing his country for the 50th time on Tuesday night, said after the 1-1 draw with Italy that Buffon was an inspiration. Speaking to FA TV, Hart said: 'I'm still learning my game and I'm still watching the likes of Buffon and the way he goes about his business at 37 years old. England and Manchester City goalkeeper Joe Hart has lavished praise on fellow goalkeeper Gianluigi Buffon. Hart has labelled Buffon, who won his 147th cap against England on Tuesday night, a 'legend of the game. 'I've got a lot more learning to do and I want to do it in this team. '[Buffon's caps total] is a long way off, but it's definitely a night to celebrate a terrific goalkeeper and a legend of the game, someone I personally look up to and it's inspirational to see. The Manchester City shot stopper, who is 10 years younger than Buffon, revealed his delight at receiving his 50th cap at the Juventus Stadium. 'I was proud of my first cap, I was proud to represent the Under 21s, and 50 caps at my age is good,' added Hart. 'I want to keep going, that's not the end for me. I just want to keep going, keep playing well for my club and country and rack them up.' **England ace Hart, pictured saving a shot by Citadin Eder, is desperate to add to his 50 international cap.**

Gold-standard summary

Joe Hart has revealed he is inspired by the likes of Gianluigi Buffon. Italy's record cap holder Buffon played his 147th game for the Azzurri. Man City shot stopper Hart recorded his 50th appearance for England. Click here to read Martin Samuel's match report from Turin.

52.5% are summarized in two sentences, and 32.6% are summarized in a single sentence. This implies that only a few PS vectors can contain 80% to 90% of a document's information from the perspective of variance preservation. We also compared the statistics with that of the human-generated gold summary. The statistics of the gold summary are presented in Figure 3. Our approach and human-generated summary abbreviated most of the document (more than 90%) to less than three sentences. Although humans can summarize a document predominantly in one or two sentences less than our model, the result is comparable, considering that

⁴<https://summari.es/>

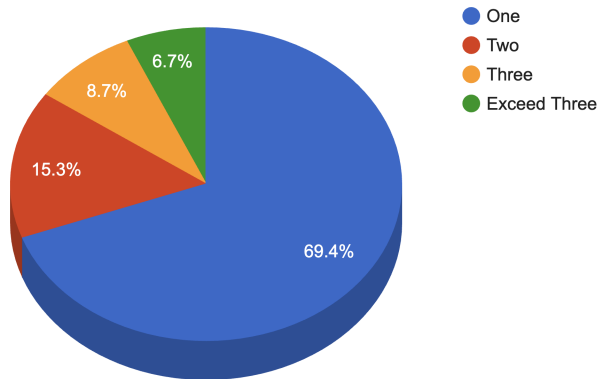


FIGURE 3. Ratio of length of selected sentences.

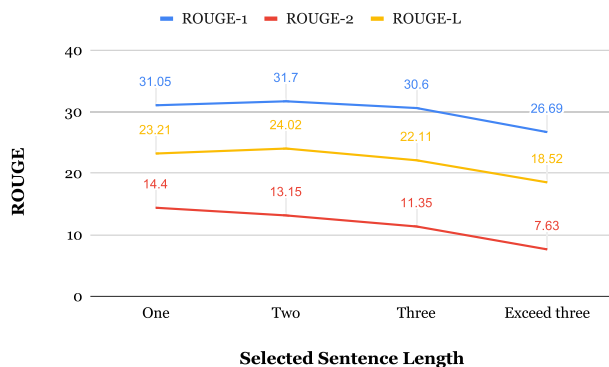


FIGURE 4. Average ROUGE score by number of selected summary sentences.

TABLE 5. Results obtained on Korean news summarization dataset.

Model	Precision	Recall
TextRank [39]	32.23	33.34
LFIP-SUM-FastText	41.15	20.61

human-generated labels are abstractive summaries that are designed to abbreviate a document into much fewer sentences.

Subsequently, we performed an analysis about the relationship between selected sentence length and the ROUGE score. Figure 4 demonstrates this relationship. The average ROUGE score decreases when the number of selected sentences exceed three. However, when the selected sentence length is three or less, which takes almost 97% of the total data, the performance is not quite sensitive to the selected sentence length. This substantiates that the number of selected sentences determined in terms of variation preservation is the number of efficient summary sentences which represent the overall meaning of a document.

E. EXPERIMENT ON KOREAN ARTICLES

One of the advantages of our approach is its applicability to low-resource languages. We experimented on the Korean news article summarization dataset.⁵ Each document has three essential sentences in the dataset as a label, annotated by human evaluators. We used Korean FastText [41] vectors as pre-trained sentence embedding. In particular, each sentence

TABLE 6. Sample results of LFIP-SUM obtained on Korean datasets. Extracted summary sentences are formatted in bold.

Example. 1

Document and extracted summary

28일 공개된 정기 공직자 재산신고에서 청와대 참모와 국무위원 상당수가 다주택자인 것으로 나타났다. 또 이번 정기 재산변동사항 신고 내역에는 요트와 보석에서 저작권까지 다양한 항목의 재산이 포함돼 눈길을 끌었다. 문 대통령을 포함해 이번에 재산을 신고한 청와대 비서관급 이상 참모진 47명의 평균 재산은 14억9천400만 원이었다. 이번 재산신고에서 가장 많은 재산을 보유한 것으로 나타난 청와대 참모는 148억6천900만 원을 신고한 주현 중소벤처비서관이다. 주 비서관의 재산은 이번에 신고된 청와대 참모진 총 재산의 5분의 1을 넘는다. 주 비서관 다음으로 많은 재산을 보유한 참모는 지난해보다 1억4천800만 원이 늘어난 54억7천600만 원의 재산을 신고한 조국 민정수석이었다. 청와대 참모 중 가장 재산이 적은 사람은 1억3천200만 원을 신고한 김혜애 기후환경비서관이었다. 청와대 참모 중 박종규 재정기획관은 지난해와 마찬가지로 자신과 배우자 명의의 서울 소재 아파트 두 채를 신고했다. 유송화 춘추관장과 강문대 사회조정비서관도 본인과 배우자, 혹은 공동 명의로 두 채의 집을 신고했다. 국무위원 중에는 강경화 외교부 장관이 서울 시내에 주택 두 채를 보유하고 있었다. 강 장관은 배우자 명의의 세일링 요트(8.55t급, 약 2천800만원)와 수상오토바이(약 400만원)도 신고했다. 주택정책을 담당하는 윤성원 청와대 국토교통비서관, 강성천 산업정책비서관, 박진규 통상비서관, 유영민 과학기술정보통신부 장관도 두 채를, 김의겸 청와대 대변인은 서울 동작구 흑석동 복합건물을 25억7천만원에 사들인 것으로 나타났다. 금과 보석류를 신고한 공직자들 역시 상당수였다. 윤강현 외교부 경제외교조정관의 경우 배우자 명의의 1천450만원 가액의 1.5캐럿 다이아몬드 반지를 신고하면서 '은혼식 배우자 선물'이라고 적었다.

Gold-standard summary

28일 공개된 정기 공직자 재산신고에서 청와대 참모와 국무위원 상당수가 다주택자인 것으로 나타났다. 또 이번 정기 재산변동사항 신고 내역에는 요트와 보석에서 저작권까지 다양한 항목의 재산이 포함돼 눈길을 끌었다. 금과 보석류를 신고한 공직자들도 역시 상당수였다.

Example. 2

Document and extracted summary

지난 4월4일 발생한 속초·고성 산불의 원인을 수사 중인 경찰이 16일이 지난 시점인 21일 한국전력 본사에 대해 압수수색을 벌였다. 이 사건을 수사 중인 고성경찰서는 이날 고성·속초 산불과 관련해 한국전력 나주 본사와 강원본부, 속초지사 등 3곳에 대해 전격 압수수색을 했다. 경찰은 오전 9시30분부터 수사관 16명을 3곳에 나눠 압수수색을 진행했다. 한전 나주 본사의 압수수색은 전신주 설치·점검·보수 등과 관련된 매뉴얼을 집중적으로 확인하고, 강원본부는 배전 운영부 컴퓨터 등의 자료를 확보 중이다. 특히 경찰이 2차 압수수색에 나선 속초지사는 고성·속초 산불의 발화지점으로 지목되는 고성군 토성면 원암리 주유소 인근 전신주를 관리하고 있다. 앞서 경찰은 지난 4월23일 한전 속초지사와 강릉지사 등 2곳에 대해 1차 압수수색을 했다. 당시 경찰은 산불 원인과 관련한 사고 전신주의 설치와 점검, 보수 내역 등 서류 일체를 압수해 분석작업을 벌였다. 이후 경찰은 지난 6월 초 고성·속초 산불의 원인으로 지목된 한전의 전신주 개폐기 유지·보수 업무와 관련해 과실 혐의가 드러난 10여명을 피의자로 입건, 이 중 4·5명에 대해 구속영장 신청을 검토한 바 있다. 검찰은 한전의 과실 책임 등에 대한 입증률 보다 명확히 하기 위해 보완이 필요하다는 입장인 것으로 전해졌다. 경찰은 “이날 압수수색은 증거 보장을 위한 것”이라고 밝혔다.

Gold-standard summary

지난 4월4일 발생한 속초·고성 산불의 원인을 수사 중인 경찰이 16일이 지난 시점인 21일 한국전력 본사에 대해 압수수색을 벌였다. 이 사건을 수사 중인 고성경찰서는 이날 고성·속초 산불과 관련해 한국전력 나주 본사와 강원본부, 속초지사 등 3곳에 대해 전격 압수수색을 했다. 경찰은 “이날 압수수색은 증거 보장을 위한 것”이라고 밝혔다.

vector is calculated as the average of the word vectors present in a sentence. As a comparison model, we used the *TextRank* method because it shares a common characteristic with our approach: both methods are fully unsupervised extractive approaches. We set the maximum number of summary sentences of *TextRank* to three because the ground truth has three sentences as a label. As an evaluation metric, we calculated the precision and recall between the predicted values and

⁵<https://dacon.io/competitions/official/235671/data/>

the ground truth. The experimental results are presented in Table 5, and two examples are presented in Table 6.

The experimental results indicate that our approach recorded higher *Precision* but lower *Recall*. This is because the LFIP-SUM model predominantly generates one or two sentences as a summary. For instance, when the model selects a correct sentence as a summary, the recall is $\frac{1}{3}$, whereas the precision is 1. In addition, we used a simple average of word vectors as a pre-trained sentence embedding model, which is a coarse method compared to those we leveraged in English datasets, such as USE and SIF. To achieve better performance, a more elaborate sentence embedding model should be implemented.

V. CONCLUSION

Text summarization is an important research area with a wide range of applications. Therefore, numerous studies have been conducted, and recently great progress has been made owing to the rapid development of deep learning. However, a large number of gold-standard summaries are required to train deep learning summarization models. This implies that it is difficult to train models for languages for which publicly available labeled corpora do not exist. Moreover, deep learning models are incapable of processing novel patterns because they concentrate on enhancing generalization performance through learning patterns from large-sized training data. However, humans with basic language understanding are able to react to novel patterns. This is an issue that should be overcome to accomplish a human-level development of artificial intelligence.

In this study, we proposed an unsupervised extractive summarization model in which parameter training is unnecessary. To achieve this, we first obtained a deep representation of documents based on pre-trained sentence vectors, positional encoding, and self-attention. Subsequently, by using PCA, we extracted PS vectors that preserve the information of a document as much as possible, and we used them to determine an appropriate number of summary sentences and to compute the importance score of each original sentence. Then, we selected the summary sentences by solving an ILP problem. Finally, we obtained the final summarization results by constructing an ensemble model based on individual models with different pre-trained sentence embedding vectors. We experimentally confirmed that the proposed model (without training examples) exhibited comparable performance to that of the state-of-the-art deep learning-based supervised model.

Our approach has the following advantages: First, the LFIP-SUM model is free from model training. This characteristic is a strong practical point for the industrial field, considering the significant amount of time and fortune consumed for building the tremendous volume of human-labeled data required. It also helps users who lack advanced computational facilities, such as GPUs and TPUs. Second, the LFIP-SUM distinguishes itself from previous approaches in that it determines the number of summary sentences dynamically. Our

method selects the number of summary sentences according to the information preservation ratio, which is a more intuitive and interpretable hyper-parameter than sentence length. In addition, through experiments, we confirmed that most of the documents are summarized within 3 sentences while preserving 80 – 90% of their intrinsic information. Finally, any sentence embedding model, regardless of the model size and language, can be jointly used with our approach. Therefore, it is possible to apply our method to low-resource languages in which human-labeled data do not exist.

However, our study has several limitations, indicating future research directions. First, the LFIP-SUM model is highly dependent on pre-trained sentence embedding methods because both sentence importance and sentence similarity are derived from the deep representation of a document. The off-chance of elaborate sentence embedding vectors prohibits our approach from achieving a promising performance, as revealed in the Korean dataset experiment. Therefore, effective sentence representation methods that can capture the intrinsic meaning of a sentence should be developed. Recent developments such as XLNet [42] and Electra [43] enabled more elaborate contextualized representations while demonstrating improved performance in several NLP tasks. Therefore, leveraging further enriched sentence representations can contribute to achieving better performance using our approach. We leave this as future work. In addition, we introduced sentence pruning for efficient calculation. Although pruning is performed for only 7% of the total test data, it causes information loss. Therefore, improvements that can reduce computational complexity while preserving information are required to enhance the practicality of our approach. Heavy computation is required to minimize the similarity score because of the y_{ij} variable—the selection indicator for sentences i and j . Therefore, reflecting the overall sentence similarity score in sentence importance $imp(s_i)$, and selected summary in descending order could address this problem to a certain extent.

ACKNOWLEDGMENT

The authors sincerely appreciate the valuable comments provided by two anonymous reviewers.

REFERENCES

- [1] K. M. Hermann, T. Kocisky, E. Grefenstette, L. Espeholt, W. Kay, M. Suleyman, and P. Blunsom, “Teaching machines to read and comprehend,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 1693–1701.
- [2] R. Nallapati, B. Zhou, C. N. D. Santos, C. Gulcehre, and B. Xiang, “Abstractive text summarization using sequence-to-sequence RNNs and beyond,” 2016, *arXiv:1602.06023*. [Online]. Available: <http://arxiv.org/abs/1602.06023>
- [3] A. See, P. J. Liu, and C. D. Manning, “Get to the point: Summarization with pointer-generator networks,” 2017, *arXiv:1704.04368*. [Online]. Available: <http://arxiv.org/abs/1704.04368>
- [4] Q. Zhou, N. Yang, F. Wei, S. Huang, M. Zhou, and T. Zhao, “Neural document summarization by jointly learning to score and select sentences,” 2018, *arXiv:1807.02305*. [Online]. Available: <http://arxiv.org/abs/1807.02305>
- [5] X. Zhang, M. Lapata, F. Wei, and M. Zhou, “Neural latent extractive document summarization,” 2018, *arXiv:1808.07187*. [Online]. Available: <http://arxiv.org/abs/1808.07187>

- [6] C. Kedzie, K. McKeown, and H. Daume, "Content selection in deep learning models of summarization," 2018, *arXiv:1810.12343*. [Online]. Available: <http://arxiv.org/abs/1810.12343>
- [7] A. M. Rush, S. Harvard, S. Chopra, and J. Weston, "A neural attention model for sentence summarization," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2015, pp. 1–11.
- [8] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 3104–3112.
- [9] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," 2014, *arXiv:1409.0473*. [Online]. Available: <http://arxiv.org/abs/1409.0473>
- [10] O. Vinyals, M. Fortunato, and N. Jaitly, "Pointer networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 2692–2700.
- [11] Z. Tu, Z. Lu, Y. Liu, X. Liu, and H. Li, "Modeling coverage for neural machine translation," 2016, *arXiv:1601.04811*. [Online]. Available: <http://arxiv.org/abs/1601.04811>
- [12] L. Liu, Y. Lu, M. Yang, Q. Qu, J. Zhu, and H. Li, "Generative adversarial network for abstractive text summarization," in *Proc. 32nd AAAI Conf. Artif. Intell.*, 2018, pp. 1–2.
- [13] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 2672–2680.
- [14] R. Kiros, Y. Zhu, R. R. Salakhutdinov, R. Zemel, R. Urtasun, A. Torralba, and S. Fidler, "Skip-thought vectors," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 3294–3302.
- [15] S. Arora, Y. Liang, and T. Ma, "A simple but tough-to-beat baseline for sentence embeddings," in *Proc. Int. Conf. Learn. Represent.*, 2017, pp. 1–16.
- [16] M. Pagliardini, P. Gupta, and M. Jaggi, "Unsupervised learning of sentence embeddings using compositional n-gram features," 2017, *arXiv:1703.02507*. [Online]. Available: <http://arxiv.org/abs/1703.02507>
- [17] F. Hill, K. Cho, and A. Korhonen, "Learning distributed representations of sentences from unlabelled data," 2016, *arXiv:1602.03483*. [Online]. Available: <http://arxiv.org/abs/1602.03483>
- [18] A. Conneau, D. Kiela, H. Schwenk, L. Barrault, and A. Bordes, "Supervised learning of universal sentence representations from natural language inference data," 2017, *arXiv:1705.02364*. [Online]. Available: <http://arxiv.org/abs/1705.02364>
- [19] D. Cer, Y. Yang, S.-Y. Kong, N. Hua, N. Limtiaco, R. S. John, N. Constant, M. Guajardo-Cespedes, S. Yuan, C. Tar, Y.-H. Sung, B. Strope, and R. Kurzweil, "Universal sentence encoder," 2018, *arXiv:1803.11175*. [Online]. Available: <http://arxiv.org/abs/1803.11175>
- [20] R. McDonald, "A study of global inference algorithms in multi-document summarization," in *Proc. Eur. Conf. Inf. Retr. Berlin, Germany: Springer*, 2007, pp. 557–564.
- [21] T. Berg-Kirkpatrick, D. Gillick, and D. Klein, "Jointly learning to extract and compress," in *Proc. 49th Annu. Meeting Assoc. Comput. Linguistics, Hum. Lang. Technol.*, vol. 1. Stroudsburg, PA, USA: Association Computational Linguistics, 2011, pp. 481–490.
- [22] K. Woodsend and M. Lapata, "Multiple aspect summarization using integer linear programming," in *Proc. Joint Conf. Empirical Methods Natural Lang. Process. Comput. Natural Lang. Learn.* Stroudsburg, PA, USA: Association Computational Linguistics, 2012, pp. 233–243.
- [23] D. Galanis, G. Lampouras, and I. Androutsopoulos, "Extractive multi-document summarization with integer linear programming and support vector regression," in *Proc. COLING*, 2012, pp. 911–926.
- [24] F. Boudin, H. Mougard, and B. Favre, "Concept-based summarization using integer linear programming: From concept pruning to multiple optimal solutions," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2015, pp. 1–6.
- [25] C.-Y. Lin, "Rouge: A package for automatic evaluation of summaries," in *Proc. Text Summarization Branches Out*, 2004, pp. 74–81.
- [26] M. Liu, L. Wang, and L. Nie, "Weibo-oriented Chinese news summarization via multi-feature combination," in *Natural Language Processing and Chinese Computing*. Cham, Switzerland: Springer, 2015, pp. 581–589.
- [27] R. Nallapati, F. Zhai, and B. Zhou, "Summarunner: A recurrent neural network based sequence model for extractive summarization of documents," in *Proc. 31st AAAI Conf. Artif. Intell.*, 2017, pp. 3075–3081.
- [28] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *Proc. 3rd Int. Conf. Learn. Represent.*, 2015, pp. 1–15.
- [29] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 5998–6008.
- [30] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," 2018, *arXiv:1810.04805*. [Online]. Available: <http://arxiv.org/abs/1810.04805>
- [31] C. Boutsidis, M. W. Mahoney, and P. Drineas, "Unsupervised feature selection for principal components analysis," in *Proc. 14th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2008, pp. 61–69.
- [32] R. Polikar, "Ensemble based systems in decision making," *IEEE Circuits Syst. Mag.*, vol. 6, no. 3, pp. 21–45, 3rd Quart., 2006.
- [33] V. Bolón-Canedo, N. Sánchez-Marroño, and A. Alonso-Betanzos, "Data classification using an ensemble of filters," *Neurocomputing*, vol. 135, pp. 13–20, Jul. 2014.
- [34] T. Boongoen and N. Iam-On, "Cluster ensembles: A survey of approaches with recent extensions and applications," *Comput. Sci. Rev.*, vol. 28, pp. 1–25, May 2018.
- [35] M. Koupae and W. Y. Wang, "WikiHow: A large scale text summarization dataset," 2018, *arXiv:1810.09305*. [Online]. Available: <http://arxiv.org/abs/1810.09305>
- [36] M. Grusky, M. Naaman, and Y. Artzi, "Newsroom: A dataset of 1.3 million summaries with diverse extractive strategies," 2018, *arXiv:1804.11283*. [Online]. Available: <http://arxiv.org/abs/1804.11283>
- [37] G. Lample, A. Conneau, L. Denoyer, and M. Ranzato, "Unsupervised machine translation using monolingual corpora only," 2017, *arXiv:1711.00043*. [Online]. Available: <http://arxiv.org/abs/1711.00043>
- [38] R. Mihalcea and P. Tarau, "Textrank: Bringing order into text," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2004, pp. 404–411.
- [39] F. Barrios, F. López, L. Argerich, and R. Wachenchauzer, "Variations of the similarity function of TextRank for automated summarization," 2016, *arXiv:1602.03606*. [Online]. Available: <http://arxiv.org/abs/1602.03606>
- [40] S. Chopra, M. Auli, and A. M. Rush, "Abstractive sentence summarization with attentive recurrent neural networks," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang. Technol.*, 2016, pp. 93–98.
- [41] E. Grave, P. Bojanowski, P. Gupta, A. Joulin, and T. Mikolov, "Learning word vectors for 157 languages," in *Proc. Int. Conf. Lang. Resour. Eval. (LREC)*, 2018, pp. 1–3.
- [42] Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. R. Salakhutdinov, and Q. V. Le, "XINet: Generalized autoregressive pretraining for language understanding," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 5753–5763.
- [43] K. Clark, M.-T. Luong, Q. V. Le, and C. D. Manning, "Electra: Pre-training text encoders as discriminators rather than generators," in *Proc. Int. Conf. Learn. Represent.*, 2019, pp. 1–18.



MYEONGJUN JANG received the B.S. and M.Eng. degrees in industrial engineering from Korea University. He is currently pursuing the D.Phil. degree with the Department of Computer Science, University of Oxford. He worked as a NLP Scientist and Engineer with the AI Center, SK Telecom, South Korea. His research interests include low-resource language learning, developing sentence embedding models, semantic analysis, and figuring out issues of applying NLP techniques to the industry.



PILSUNG KANG received the B.S. and Ph.D. degrees in industrial engineering from Seoul National University. He is currently an Associate Professor with the School of Industrial Management Engineering, Korea University, South Korea. His main research interests include developing machine learning algorithms for both structured data and unstructured data (image, video, and text) and applying them to solve engineering and business problems, such as fault classification in manufacturing, abnormal behavior detection from system logs, and sentiment classification from news and review texts.

...