Topic modeling based Siamese-LSTM-BERT model for Semantic Document Similarity

Donguk Kim

The Graduate School
Yonsei University
Department of Industrial Engineering

Topic modeling based Siamese-LSTM-BERT model for Semantic Document Similarity

A Master's Thesis
Submitted to the Department of Industrial Engineering
and the Graduate School of Yonsei University
in partial fulfillment of the
requirements for the degree of
Master of Industrial Engineering

Donguk Kim

December 2021

This certifies that the Master's Thesis
of Donguk Kim is approved.

Wooju Kim

Thesis Supervisor : Wooju Kim

Thesis Committee Member : Chang Ouk Kim

Thesis Committee Member : JuneSeok Hong

The Graduate School
Yonsei University
December 2021

Acknowledgement

This work used AI Data provided by AIHub. AI data is the result of the National Information Society Agency (NIA) business and complies with the NIA's required utilization policy.

This work used Korean BERT pre-trained cased (KoBERT) provided by SKTBrain and complied with the requirements of Apache License 2.0.

Table of Contents

List of Tables

List of Figures

Abstract

Topic modeling based Siamese-LSTM-BERT model for Semantic Document Similarity

Donguk Kim

Department of Industrial Engineering

The Graduate School

Yonsei University

BERT shows the state of art in various natural language processing tasks and generates text embedding of rich expressions. However, since the length of text applicable to BERT is limited, research has been mainly conducted only on short sentences. There have been several attempts to generate BERT-based document embedding, but embeddings were generated only with a part of the document by extracting the contents of the document. Since document extraction methods cause loss of information on documents, there is a limit to forming accurate document embedding. The method of using document excerpt information requires empirical knowledge of the document structure or conditions for knowing important parts of the document in advance. However, for documents for a particular domain, lack of professional domain knowledge makes it difficult to know important content in the document, and whenever the type of document differs, the location of the document to be embedded must be changed every time, and inaccurate document embedding may be generated.

This study proposes a document embedding method for long documents based on BERT. The model proposed in the study divides the document into segments and regards it as each sequence state. This sequence can be expressed as a single document embedding through LSTM and can help use the document's overall information. In addition, to generate document embedding suitable for the domain, topic distribution information was combined for each segment using topic modeling to generate document embedding specific to the domain.

The proposed model uses the Siamese Network to determine the similarity between documents based on document embedding that combines topic distribution information and segments. In addition, it improves the maximum applicable length problem of existing BERT so that embeddings can be generated based on global information of documents instead of using only part of documents for embedding and combines topic distribution information with document embedding to show better results than existing methodologies.

Keywords : BERT, Topic Modeling, Siamese Network, Semantic Similarity, Document Embedding

1. Introduction

Semantic textual similarity is one of the natural language tasks that determine whether they are similar or not on the two texts. In the existing semantic textual similarity, task is being conducted on short texts such as sentences. This task takes relatively little efforts and resources because it is relatively short compared to long text. However, in real world, a lot of long text data is generated, and judging long documents directly is a task that requires a lot of efforts and resources, unlike sentences. In particular, in the case of documents limited to laws, patents, and medical fields where professional domain knowledge is used, there are considerable difficulties for non-professionals to determine the similarity between documents. This is because documents belonging to a specific domain are not only domain-specific terms but also very different ways of describing sentences. In general, these tasks are determined by experts, and a lot of costs are incurred in this process, and additional results of the determination work have a problem that they have no choice but to rely on experts.

To solve this problem, a similarity matching work based on appropriate document embedding is required. Traditional document embedding methods mainly used count and frequency-based document presentation methods and keyword-based algorithms were used. This method is very intuitive and easy to access by expressing a document based on the frequency of specific words present in the document. In the case of the TF-IDF, it is one of the traditional documents embedding representation that can identify the importance and unique degree of words by measuring the frequency of words in a document (Jones, K.,

1972) and using n-gram in sentences is also an another traditional way to embedding texts to more richer representation (Zhang, X. et al., 2015). However, in the case of such document embedding, the keyword of the document may be reflected, but the semantics of the word may vary depending on the context, so there is a limit to how the entire document is embedded.

Based on the neural network, many document embedding models including the meaning of sentences have been worked. CNN-based (Yoon Kim., 2014) or BiLSTM-based document embedding contains the information of each sentence component in learning while preserving the information of the emerging position of the sentence. Through the convolution natural network, it is possible to effectively express more compressed information and n-gram words in existing embeddings by adjusting the size of the window as well as location information of the text.

The hierarchical attention networks (Yang, Z. et al., 2016; Abreu, J. et al., 2019), which applies self-attention (Vaswani, A. et al., 2017) to identify important words and sentences within a document using document form, a hierarchical structure, is one of the representative models for document classification. The key idea of using a hierarchical model is that documents are the result of sequences of multiple sentences, and sentences are also the result of sequences of multiple words. Documents should not be judged simply by the frequency of words, and the characteristic that documents are constructed by sentences composed of specific word expressions to properly express the document was utilized. However, existing models that only rely on LSTM show excellent performance in document classification. When we apply LSTM, if text length is long, it takes a very long time to learn and has limitations in generating accurate document embedding.

BERT (Devlin, J. et al., 2018), based on the transformer structure, improved to identify the meaning of words that may vary depending on the context, and showed performance in achieving state of the art in various natural language processing tasks. Task is being conducted using BERT-based embedding in many natural language processing problems, and semantic textual similarity is also one of them. However, BERT has a fixed maximum limit size of 512 tokens in which text can be encoded, so many studies have been limited to short sentences. BERT uses self-attention after text is encoded, so the longer the length, the longer it takes, and for this reason, even if the length of the text is not long, previous studies have only used some data with sufficient length of text applied to BERT to reduce the speed of reference.

There was work to embed long documents, but the document length consists of datasets smaller than the maximum length size or if exceeds the maximum length, a method of creating document embedding by truncated document (Adhikari, A. et al., 2019). The position in which the document is truncated is also variable. In general, documents are truncated from the head and used for embedding, but more test errors may occur than when documents are truncated only from the tail or truncated from the head and tail and mix them (Sun, C. et al., 2019). This is method of generating embeddings by truncated only a part of a document requires empirical knowledge that the user must be empirically aware of which part is important within the document or how the document is structured. This means that experts with domain knowledge are needed, so it can be said that it is difficult to reduce a lot of efforts and costs. Most importantly, even if we know what part is key in a document and generate a document embedding, it always a complicated situation that all types of documents must be different and changed and embedding the entire document with only some information cannot avoid loss of information.

In this study, we propose a new embedding method and semantic document similarity model by improving the existing BERT-based document embedding method. First, a segment-based document embedding method that uses all document information regardless of the length of the document is used. All documents can be represented throughout the document using a sequence of information without truncated and using only part of the document. This approach can minimize document information loss by using document global information, unlike conventional methods using document local information. Second, a document embedding method specialized in the domain is used using topic modeling. At the same time as embedding the entire document, topic information of the document is added so that the domain characteristics of the document can be well reflected. Finally, document embedding with topic information is used in the siamese network method to determine similarity between documents. In the news data and the Supreme Court of Korea judgement data, the proposed method showed higher performance than the existing methods.

2. Related work

2.1 Siamese Network

The Manhattan-LSTM (Mueller, J. et al., 2016) is a semantic textual similarity model between sentences based on siamese network. Embedding is performed for each word, and the sequence of these embeddings is finally generated for each text through the LSTM. The difference between the embedding vectors is calculated based on the Manhattan distance to determine the similarity between the two texts based on the distance, and a siamese network structure is formed using a shared LSTM in the model training process (Figure 1). LSTM is used to utilize the sequence of sentences, but if the sequence lengths is long, the time required for model training is too much, and the performance is poor.

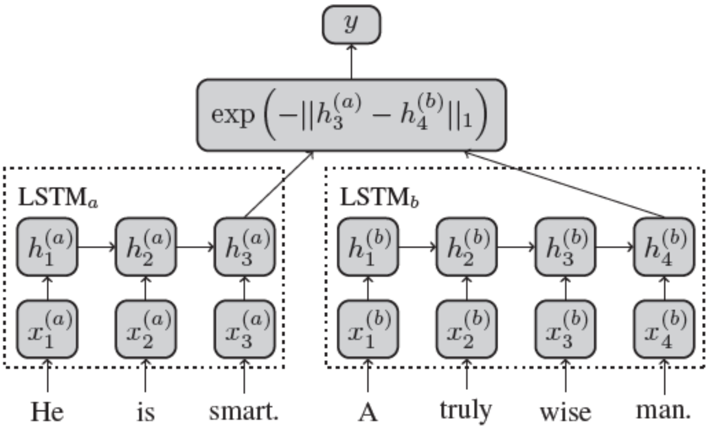

Figure 1. Manhattan-LSTM

The Dependcy-based LSTM (Zhu, W. et al., 2018) is a another siamese network-based model that determines the similarity between sentences by combining support component that help sentence embedding by adding components of sentences. All elements of a sentence are used, but the elements between the two sentences can be compared by additionally combining the subject and object information that is considered the most important in the sentence with an embedding vector to generate richer sentence embedding than the existing sentence (Figure 2). This work can help better reflect the characteristics of each sentence by adding weights to more important factors even within the existing sentence. However, like the Manhattan-LSTM model, there is a problem that the learning time increases when the sequence is long.

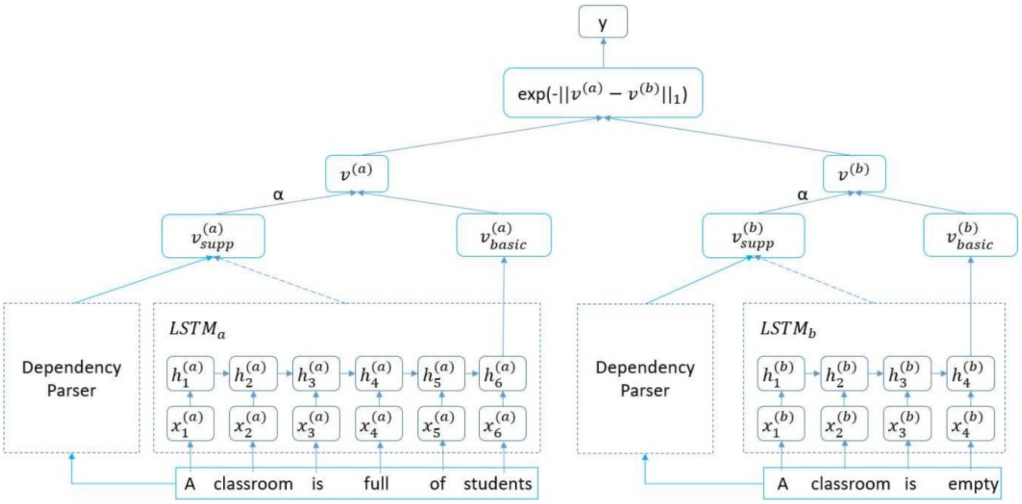
y
exp(-||$v^{(a)} - v^{(b)}$||$_1$)
$v^{(a)}$
$v^{(b)}$
α
α
$v^{(a)}_{supp}$
$v^{(a)}_{basic}$
$v^{(b)}_{supp}$
$v^{(b)}_{basic}$
Dependency Parser
$LSTM_a$
$h_1^{(a)}$ $h_2^{(a)}$ $h_3^{(a)}$ $h_4^{(a)}$ $h_5^{(a)}$ $h_6^{(a)}$
$x_1^{(a)}$ $x_2^{(a)}$ $x_3^{(a)}$ $x_4^{(a)}$ $x_5^{(a)}$ $x_6^{(a)}$
A classroom is full of students
Dependency Parser
$LSTM_b$
$h_1^{(b)}$ $h_2^{(b)}$ $h_3^{(b)}$ $h_4^{(b)}$
$x_1^{(b)}$ $x_2^{(b)}$ $x_3^{(b)}$ $x_4^{(b)}$
A classroom is empty

Figure 2. Dependcy-based LSTM

2.2 BERT

BERT (Bidirectional Embedding Representations from Transformers) is a language model that utilizes only the encoder part of the transformer. BERT pre-trained MLM (Masked Language Model) and NSP (Next Sentence Prediction) tasks through unsupervised learning and recorded the state of the art in various NLP tasks and fine-tuning for each task is possible. Text embedding uses tokenizer in BERT to separate the text into tokens and performs encoding by combining token embedding, segment embedding, and position embedding information for each token (Figure 3). Text embedding is utilized through the pooling operation of BERT CLS token or BERT embedding pooling. However, all these methods have a maximum limit length of 512 tokens, which is a limit in the process of embedding long sequences at once, and many studies use the method of truncating text to improve this.

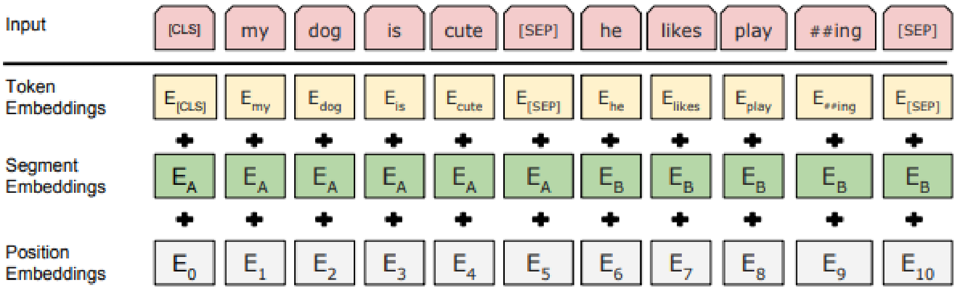

Figure 3. BERT input representation

2.3 BERT with Topic Modeling

Several tasks have been done to generate text embedding by combining domain information with BERT to conduct the classification and relationship of data in a specific field. Probability estimation was performed based on topics to generate domain information, and LDA (Latent Dirichlet Allocation) is a representative topic modeling (Blei, D. et al., 2003). LDA is a method of assuming that documents consist of topics and that topics generate words based on probability distribution. This probability distribution can be used to infer the topic distribution for documents, and it can be applied not only to documents but also to sentences.

In addition to topic modeling, author information and metadata embedding were concatenated with BERT to enrich information on book embedding. In this way, there is a method of combining with the embedding value of BERT using other additional domain information along with BERT (Ostendorff, M. et al., 2019). The tBERT (Peinelt, N. et al., 2020) is a model that determines the similarity between sentences belonging to a specific domain using topic information. The CLS token obtained by adding two sentences to the BERT model and the topic information vector in the two sentences are combined to finally determine the similarity between the two sentences (Figure 4). To identify topic information on sentences, the probability of topic distribution for sentences can be trained through LDA topic modeling. For semantic textual similarity, topic distribution vectors can be inferred through LDA using sentences. The domain characteristics of the sentences are reflected by adding a specific topic distribution of the sentences. However, in the process of reflecting topic distribution information, all sentence information with the same prefix as the survey, not the specific morpheme, is used, so the performance in topic modeling is not clearly reflected.

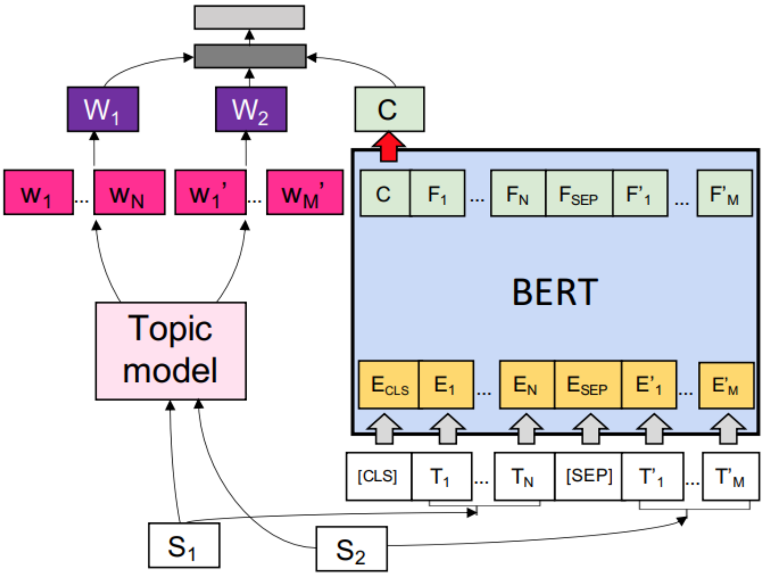
W1
W2
C
w1 ... wN
w1' ... wM'
C F1 ... FN FSEP F'1 ... F'M
BERT
Topic model
ECLS E1 ... EN ESEP E'1 ... E'M
[CLS] T1 ... TN [SEP] T'1 ... T'M
S1
S2

Figure 4. tBERT

2.4 Sentence-BERT

The Sentence-BERT (Reimers, N. et al., 2019) applied the siamese network structure to the BERT (Figure 5), allowing it to determine the similarity of the large-scale dataset that the BERT could not proceed. Unlike the use of only CLS token in the existing BERT embedding, Sentence-BERT used the vectorized embedding by pooling the BERT embedding. In this study, based on BERT, similarity determination between sentences was performed through comparison between embedding and embedding of existing language models. Instead of calculating the manhattan distance used in Manhattan-LSTM, the calculation between embeddings is performed and the similarity is measured through softmax and cosine similarity. These text embeddings combined differences between vectors and used to determine similarity to show high performance. However, in the case of this study, as in the past, it has been limited to only sentence embedding and the limit of the maximum length has not been solved like BERT.

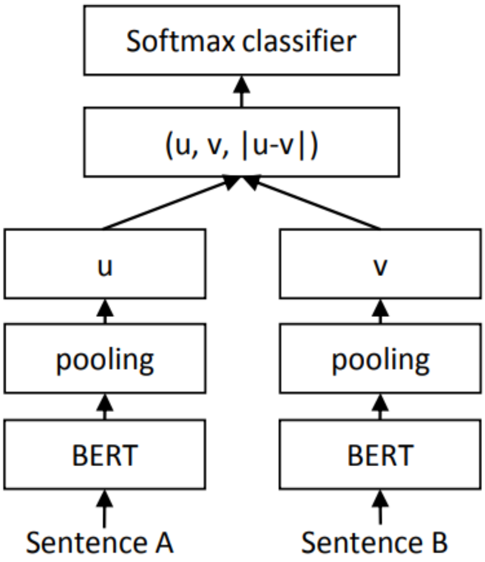

Figure 5. Sentence-BERT

2.5 Document Variants Embedding

Document variants has conducted to improve the maximum length text of BERT. The length of the text applied to the BERT is 512 tokens, and if the length of the document is longer than this, variants work was performed on the document (Joshi, M. et al., 2019). Segment overlapping using interpolation was used, but there is a limitation in that half of the document data used, less than 512-length, was mixed and used, resulting in the embedding of long documents under the influence of short documents.

In general, document truncation was performed, and according to the number of chunks being truncated (Wan, L. et al., 2019), documents were embedded in Figure 6, or documents were divided into segments and overlapped (Pappagari, R. et al., 2019). There is no exact criterion for the length of overlap, and each data has a limitation in that the length of overlap and the window size are different. Document variants are generally limited to cutting or overlapping documents, creating document embedding using BERT.

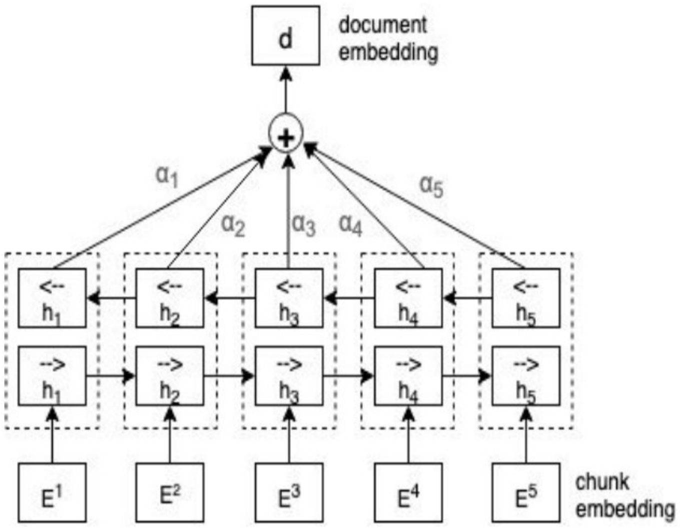

Figure 6. Document embedding through chunk

3. Topic modeling based Siamese-LSTM-BERT

The topic modeling-based Siamese-LSTM-BERT model in Figure 7 contains the topic information of the document and determines the similarity by generating an embedding based on the BERT CLS token sequence throughout the document. In the siamese network structure, the BERT and LSTM share weights, and the entire model consists of three parts : BERT-Encoder, Segment-Level Topic Inference, and Document Similarity Comparison. Two documents can proceed with three processes and then generate embeddings u, v of individual documents. After performing vector operations between document embedding, results for semantic document similarity can be obtained through softmax.

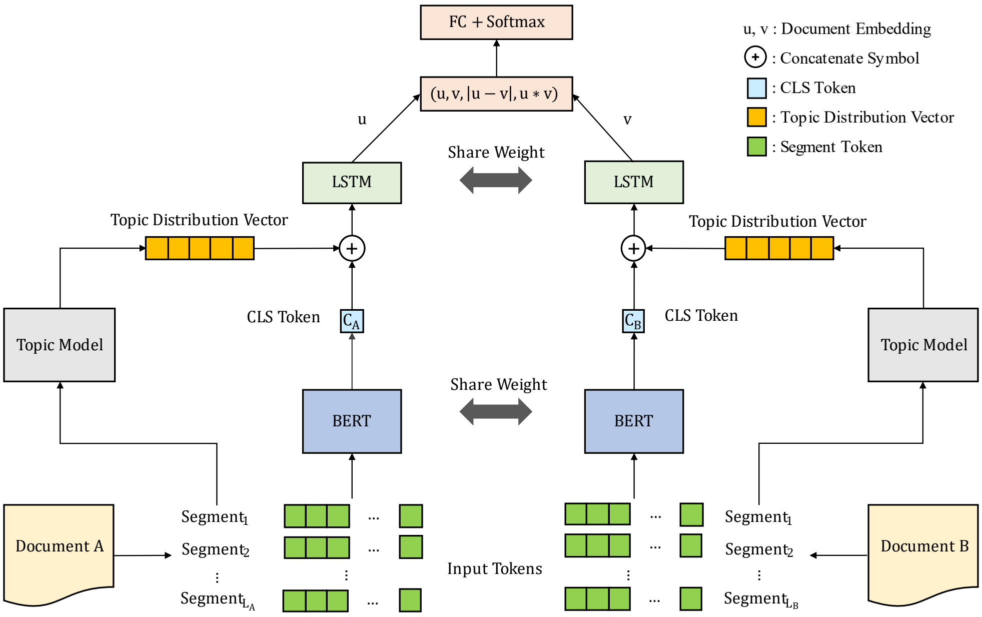

Figure 7. Topic modeling based Siamese-LSTM-BERT

3.1 BERT-Encoder

The BERT- Encoder is a part of document embedding based on the BERT, and proceeds in the order of the document sequence segmentation process and the BERT for segmentation presentation. BERT used koBERT pre-trained by SKTBrain. The koBERT is a model trained in 5M sentences and 54M words on the Korean Wiki. Korean text has limitations in creating tokens based on spacing. This is because Korean text is not properly followed by spacing rules and has a linguistic characteristic of agglutinative language. The function of the word is determined by the word root and affix. For example, "나는", "나를" and "나도" have different meanings depending on the types of combined postposition. Since Korean proceeds with tokenization based on morpheme, if token embedding is performed based on tokenization in other languages, the length of the token sequence becomes excessively long. Fine-tuning was performed using koBERT to apply linguistic characteristics and minimize the length of the sequence.

The document sequence segmentation part divides the document into L segments according to the maximum length of the BERT. L is defined as the number of segments and is a variable because different segments occur for each length of the document. In Figure 8, document A and document B have different lengths L_A, L_B. The divided segment consists of 512 tokens length. Based on the segment, the fine-tuning of the BERT is conducted, and CLS token generated by this means segment regeneration.

The CLS token information on the segment is Eq. (1). Each CLS token is C and *d* has a value of 768, which is the hidden layer size of BERT. The method of generating embeddings by dividing documents in a segmental manner in the BERT-Encoder is not limited to local information that uses only part of previously used documents, but global information of documents can be used.

$$\mathrm{C} \in \mathrm{R}^{d} \quad (d = 768, \text{BERT hidden layer size}) \tag{1}$$

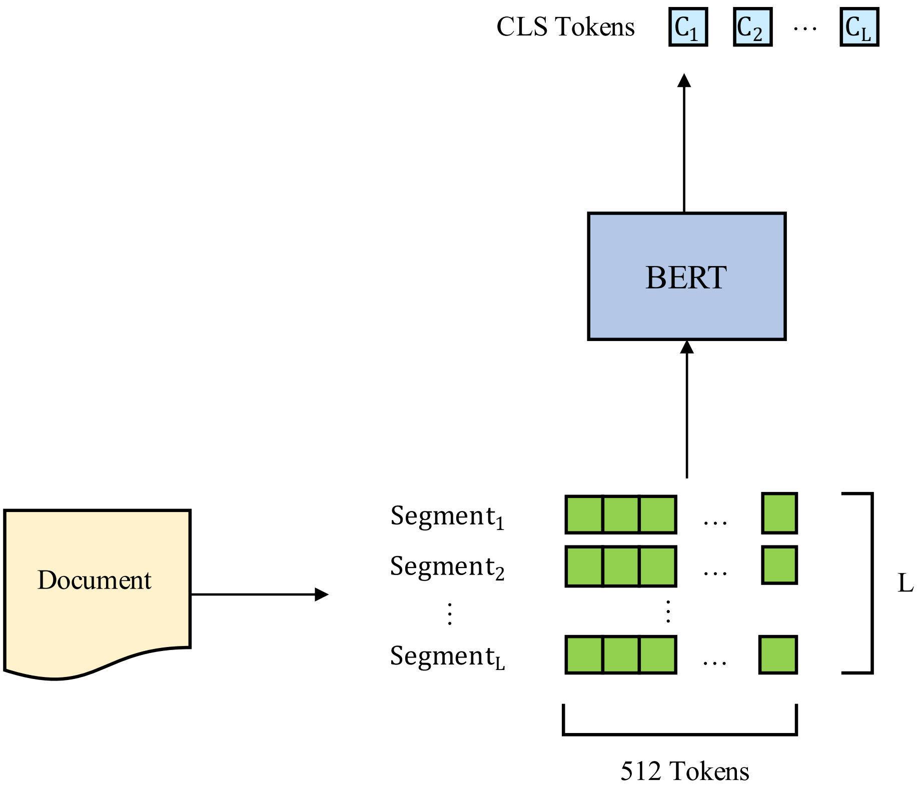

Figure 8. Document sequence segmentation and segment representation

3.2 Segment Topic Inference

In the process of the topic model, segment-level topic distribution inference is performed. The LDA topic model is constructed based on a vocabulary consisting only of the nouns. In addition to common nouns, the noun vocabulary includes compound nouns in the form of n-gram. In Korean text, the meaning of compound nouns may be very different by spacing, so a noun vocabulary built in advance was used when training the model so that the topic model can accurately infer them. In topic model matrix (Figure 9), it is possible to identify the word distribution and probability values for each topic of the trained model. The row of the matrix is the size of the vocabulary, and the column is the number of topics.

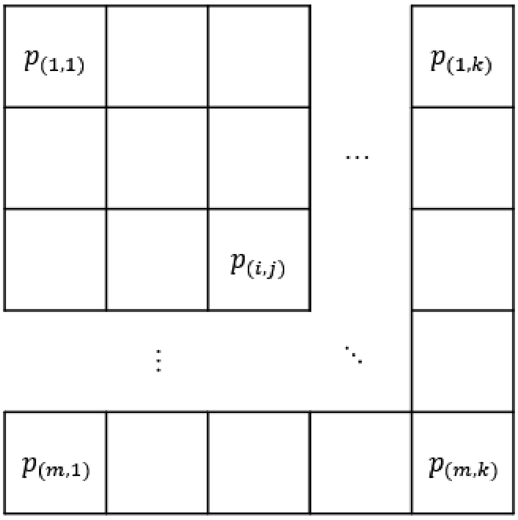

Figure 9. Topic model probability matrix

The sum of the probability values of words appearing in a topic with a specific subject is all 1. In Eq. (2), the total sum of the probabilities of word distribution by all topics is k.

$$\sum_{j=0}^{k}\sum_{i=0}^{m} p_{(i,j)} = k \qquad (2)$$

Each segment can derive L of the segment's topic distribution by the topic model. At this time, TD is the topic distribution vector. Through the topic model, it is possible to infer $\mathrm{TD_L}$ for each segment information on what topic distribution segment forms in the document (Figure 10). In Eq. (3), the topic distribution vector has a vector dimension by k, which is the number of topics. The number of topics applies differently to each type of document data to be used. Just as the document is divided into segments and viewed as a single sequence, the topic distribution information of segments can also be regarded as a single sequence.

$$\mathrm{TD} \in R^{k} \quad (\text{k = the number of topics}) \qquad (3)$$

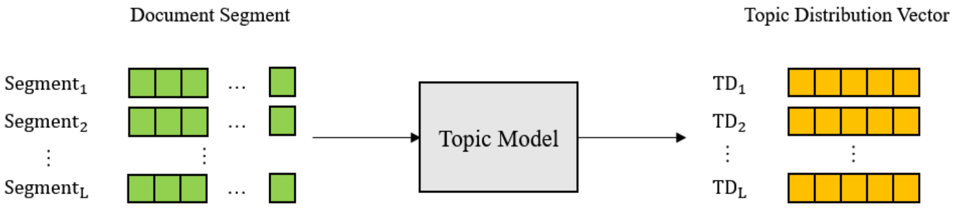

Figure 10. Segment topic distribution inference

3.3 Document Similarity Comparison

At the end of the model, segment-level presentation with topic information generates embeddings that contain the topic information of the document and determines the similarity of the document using Siamese-LSTM. Concatenation is performed for each segment of the CLS token generated by BERT encoder and the topic distribution vector generated by the topic model (Figure 11).

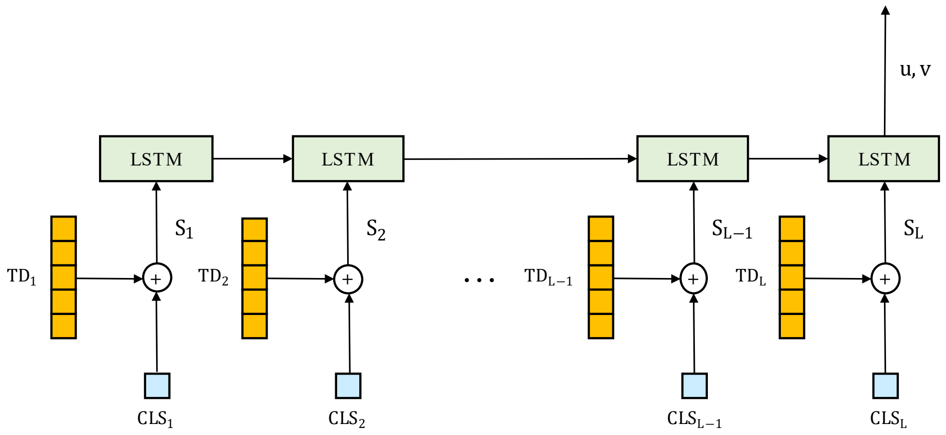

Figure 11. Segment-level representation with topic information

The S_t in Eq. (4) generated by vector concatenation is defined as segment-level presentation with topic information and consist of Eq. (5). By sequentially connecting the sequence of S_t using LSTM (Figure 11), the embedding expression of the entire document generating the last hidden state, topic information can be contained. Using LSTM, the process of connecting each sequence of dividing the document into segments in the initial process is performed. In the process of connecting, not only the existing document information but also the topic information in the document continues. The document is also a sequence of sentences, but can be thought of as a sequence of topics. Because the segmented pieces are combined, the maximum limit length of the BERT can be overcome and any long document can be generated through the proposed method.

$$\mathrm{S_t} = [\mathrm{C}_t\ ;\ \mathrm{TD_t}]\ \text{(; is concatenated symbol)} \tag{4}$$

$$\mathrm{S_t} \in R^{d+k} \tag{5}$$

Since the LSTM is composed of siamese network, each document embedding u and v with shared weight can be generated, and even if the length of the document is different, it has the same dimension. Additional vector connection is performed through two document embeddings and each difference and doc multiply, and then the similarity between the two documents is finally determined through fully connected layer and softmax function.

4. Experiments

4.1 Datasets

The experimental data to verify the model, news data built by AIHub for reading comprehension and the Supreme Court of Korea judgement data specialized in the legal domains were used. In the Supreme Court judgment dataset, the contents of the document were extracted and used only the contents of the judgment described in detail in the case. In the case of other lists, the content is often omitted, and since the entire content of the judgment is summarized, the length of the document is short, so it is more challenging to make a similar judgment only with the content of the judgment.

Table 1. Legal document example

Contents of Judgment	Case Label
채권자가 사해행위의 취소와 함께 수익자 또는 전득자로부터 책임재산의 회복을 구하는 사해행위취소의 소를 제기한 경우 그 취소의 효과는 채권자와 수익자...	민사
이 사건 등록서비스표의 지정서비스는 피고의 선사용서비스표들의 사용서비스와는 그 서비스의 제공 목적, 서비스의 성질 및 내용 등이 상이하여 전혀 이질적...	특허
피고인들은 위 삼원산업을 퇴사하여 동종 회사인 '(상호 생략)업체'를 설립하여 운영해 나가되 위 삼원산업의 선박용 엔진부품을 훔쳐 새로 설립하는 회사에...	형사

Since the model experiment is not a single document classification but a measure on the similarity between documents, the document pair was reconstructed based on the label of news data and judgment data. The news data used 9 categories, such as sports, economy, and politics, and the judgment data referred to 6 case-type labels, such as civil, patents, and criminal case (Table 1).

Table 2. Dataset statistics

	N	L (Avg)	L (Max)	L (Min)
News	456,652	1,704	5,826	844
Legal	1,226,032	5,609	9,501	3,887

The label of the document pair consists of a binary label with a label of positive meaning similar if the two categories or event types are the same, and a label of negative if not. In the case of existing studies, data of text length applicable to BERT were mixed and used without additional document modification. However, since short documents cannot be accurately measured for long documents, only long documents were sampled. N is denoted as the data sample, and L is denoting as the length per document. The document length for each data consists of an average of about 1,700 and 5,600 lengths (Table 2). Datasets are sampled to exceed the BERT maximum limit length of 512 tokens.

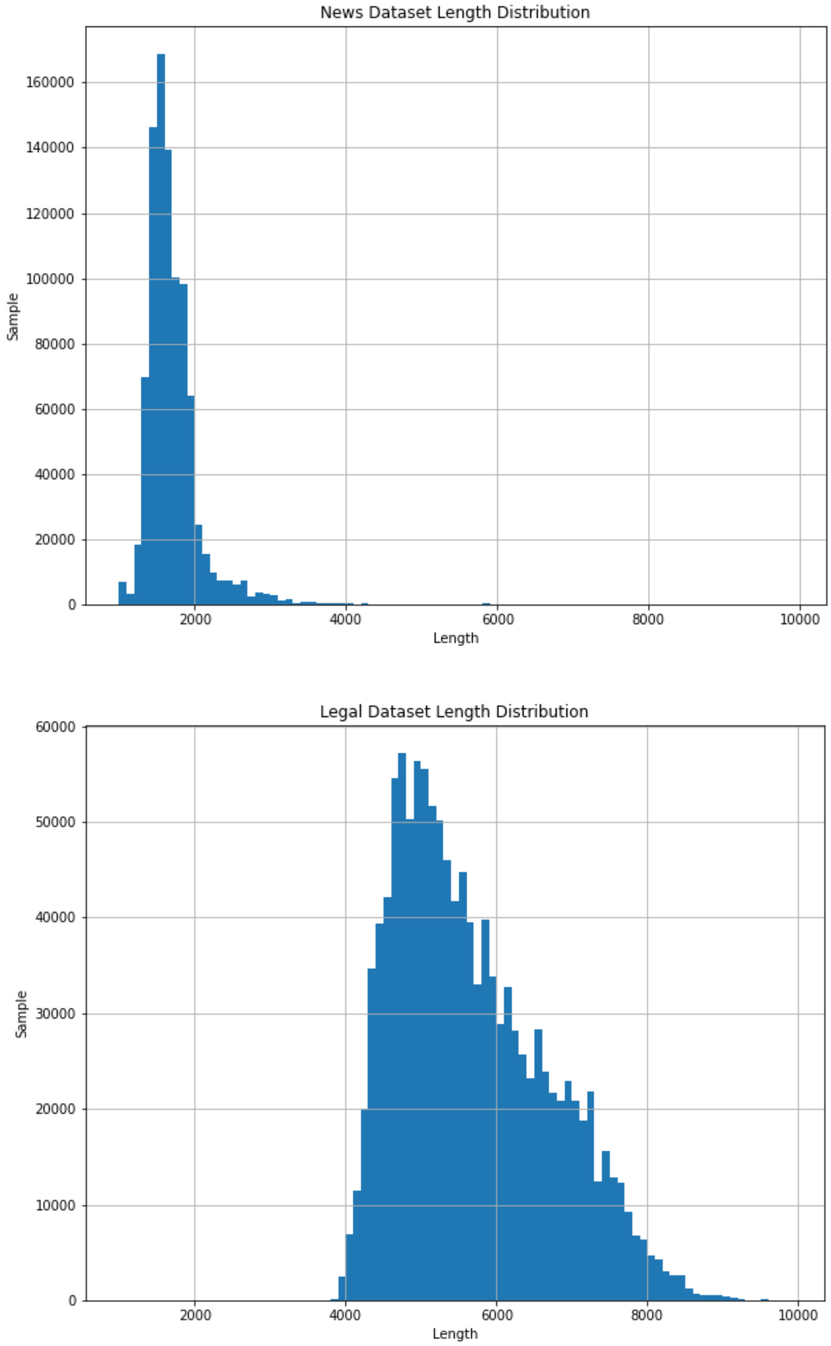

Figure 12. Datasets length distribution

To measure the effectiveness of performance evaluation for long domain-specific documents, legal domain data was approximately three times more augmented than the number of samples of news data, and the length of documents was also increased. The average length of legal data is 5,600, which is close to the maximum length of news data, and most samples consist of documents that are much longer than news (Figure 12). Data balancing of labels was set to 1:1 to train the model evenly. In the configuration of the evaluation data, train, valid, and test were divided into 6:2:2, respectively, and all pairs of documents were separated independently in each dataset so that data did not duplicate.

4.2 Model Settings

The model was divided into two parts and the training was conducted. In the first part, fine-tuning was performed for BERT. In the fine-tuning, a softmax function was added to the layer on which BERT CLS token is output, the same as the existing document classification operation, so that classification tasks for each data were performed. Through this fine-tuning, when a document enters BERT, it is possible to obtain a CLS token value that allows the label of the document to be properly inferred. In the second model part, vector calculation of LSTM and document embedding and training softmax function are performed to discriminate similar documents. Our model is trained so that BERT and LSTM share the same weight. While setting weight to be shared, it was possible for the document pair to derive result values for the same similar discrimination regardless of order.

Table 3. Model hyperparameters

	Epoch	Batch Size	Optimizer	Learning Rate
BERT	5	8	AdamW	5e-5
LSTM	10	64	SGD	0.01

When koBERT fine-tuning was performed, the input size of the text was set to 512 tokens, epoch of 5, batch size of 8, optimizer of AdamW, and learning rate of 5e-5. LSTM is trained epoch of 10, batch size of 64, optimizer of SGD, and learning rate of 0.01. The hyperparameters are presented in Table 3.

4.3 Topic Modeling

We trained topic modeling by extracting only nouns individually to improve the limitations that we proceeded by constructing topic modeling, including other insignificant components including nouns among the existing sentence components. We used the mecab tokenizer to build the dictionary required for topic modeling and extracted words tagged into noun series such as NNG and NNP. After extracting nouns, a total of 15,432 words dictionaries in news data and 17,147 words dictionaries in legal data were generated by manually removing too few or too many words.

The training of LDA topic modeling was conducted by increasing the optimal topic number by 10, and the evaluation metrics according to training used the language score.

Coherence score (Röder, M. et al., 2015) was used as the first indicator, and perplexity score (Chang, J. et al., 2009) was used as the second indicator. Coherence score is an evaluation index that measures consistency with the subject and measures the degree to which it is suitable for real people to interpret Eq. (6). When determining the optimal number of topics, the topic distribution according to the number of topics and the corresponding probability must be properly configured to infer the exact topic distribution of the document segment. Perplexity score measures the complexity of the language model and is an evaluation index of whether the topic model can predict the observed values Eq. (7).

$$C_{UMASS} = \frac{2}{N(N-1)} \sum_{i=2}^{N} \sum_{j=1}^{i-1} log \frac{P(w_i, w_j) + \varepsilon}{P(w_j)} \tag{6}$$

$$Perpelxity = 2^{H(p)} \tag{7}$$

Since we are more interested in whether the topic probability distribution of segments in the document is well inferred than the performance of the language model itself, we set the optimal number of topics with an emphasis on coherence score indicators Based on the coherence score and perplexity score, we trained LDA by setting topics optimal local option sections in coherence value as the number of topics and this point has a good score on the perpelxity score. The number of topics was set at 20 in news data (Figure 13) and 90 in legal data (Figure 14).

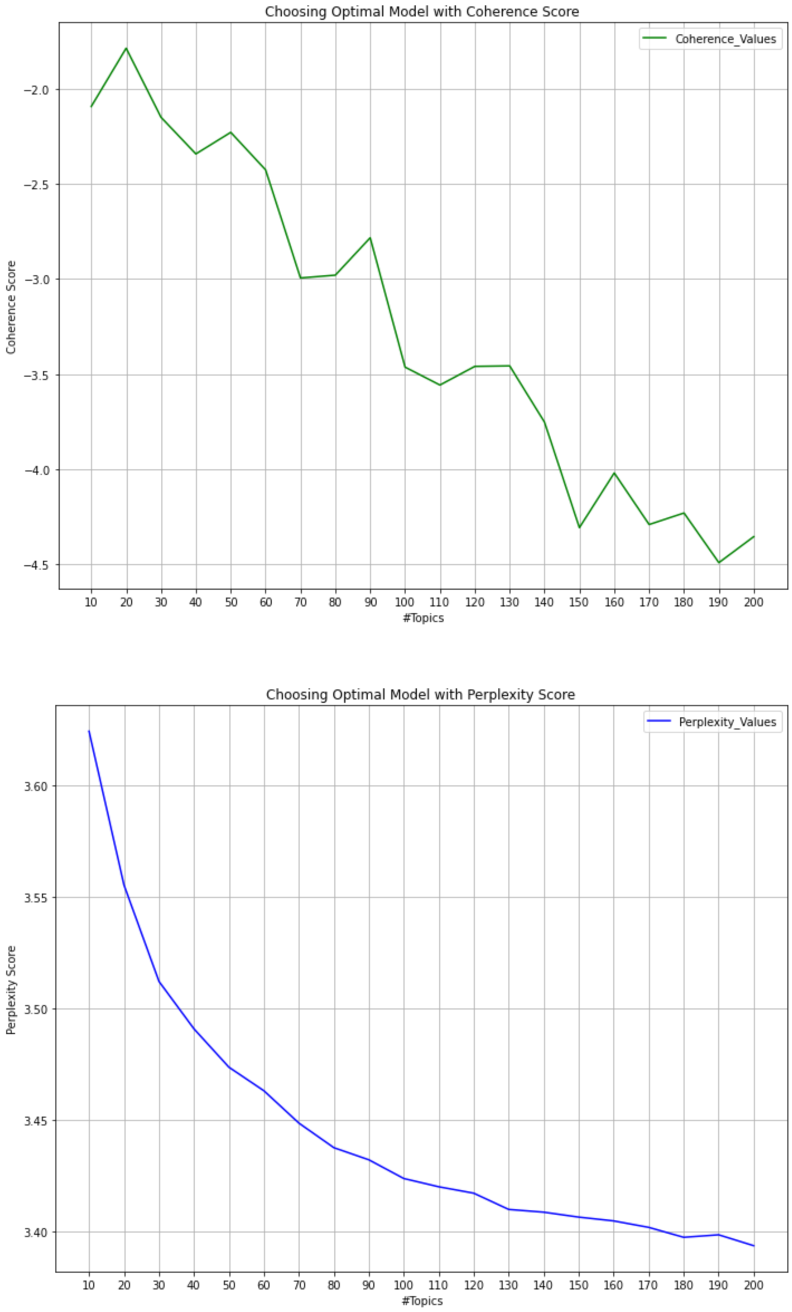

Figure 13. Optimizing the number of topics in news data

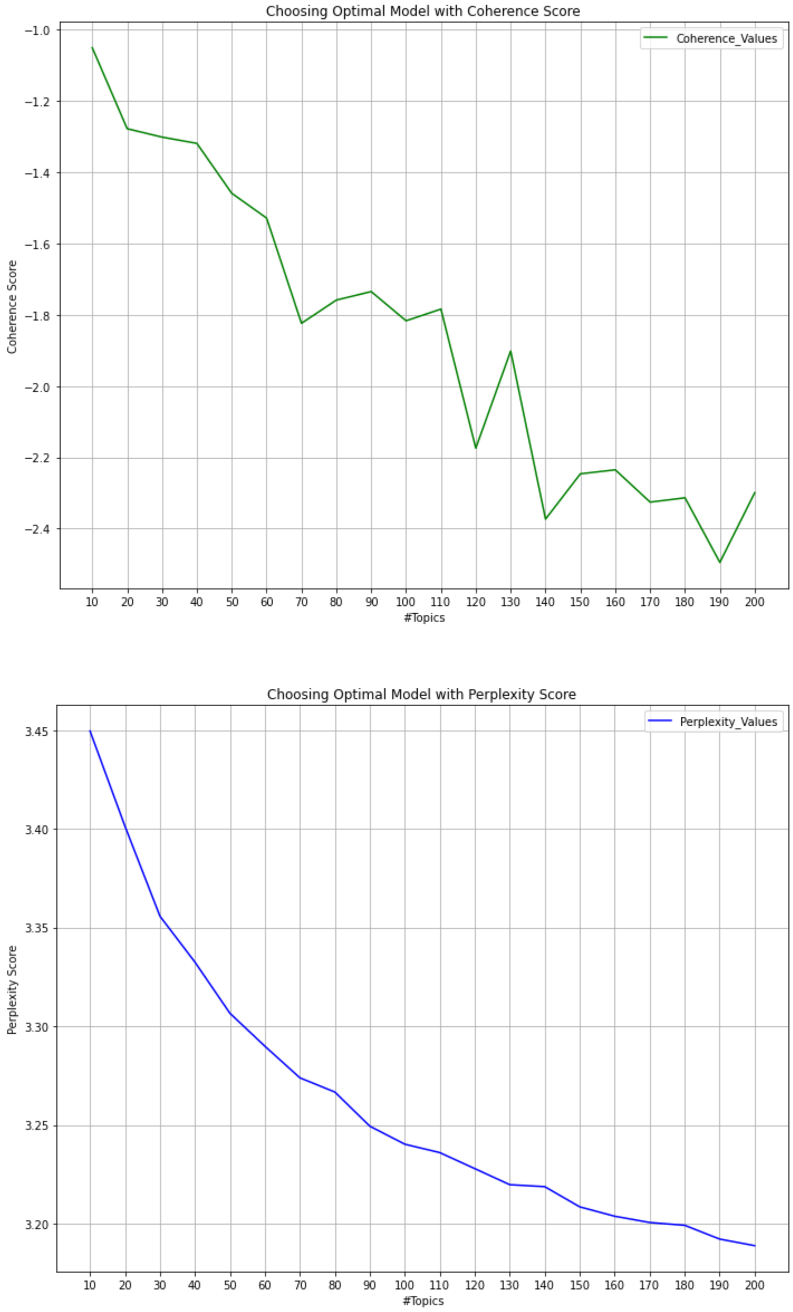

Figure 14. Optimizing the number of topics in legal data

4.4 Results

In this study, four experiments are conducted. The first experiment conducted a performance comparison experiment on document embedding itself by applying document embedding of traditional document embedding models and BERT-based models to generate document embedding in our proposed model and part of the model that performs vector computation and document-like discrimination. The second experiment was conducted on existing similar discrimination models, not document embedding. In the third experiment, performance evaluation was performed according to the vector concatenation strategy between document embedding before applying the softmax function. In the last experiment, a time comparison experiment was conducted to infer CLS token by document type. The baseline model was experimented with a technique using traditional word counts or frequencies, topic modeling, and a model based on BERT embedding.

In the experiment, the data form of the model was considered. The document types are divided into four types, and all (A) when documents are used as they are, truncate (T) when documents are randomly cut and used, overlap (O) when documents are overlapped, and segment (S) for the segment method we proposed, and the performance was compared accordingly. The document type is a measure of the amount of information for maximizing the document information used.

Table 4. Comparison of results document embedding

	Methods	Document Type	Accuracy (%)	
			News	Legal
Baseline Model	TF-IDF	A	56.20	82.81
	Topic Modeling (LDA)	A	61.31	80.12
	Doc2Vec	A	62.02	82.92
	BERT CLS-vector (CLS)	T	71.68	86.15
	Avg. BERT-Embedding	T	72.29	86.63
	Max. BERT-Embedding	T	72.49	86.86
	tBERT	T	72.79	86.13
	BERT(CLS + P_{CLS}) + LSTM	O	74.05	87.97
Proposed Model	BERT (CLS) + LSTM	S	76.57	93.81
	BERT (CLS) + LSTM + LDA	S	**78.54**	**94.20**

Table 4 shows better performance even when compared to the way the entire document is used or only the document part is used. The method of combine topic information into components with document embedding shows better accuracy than the overlapping approach that combines CLS token and softmax function probability (P_{CLS}) values for classification.

Table 5. Comparison of results semantic textual similarity model

	Methods	Document Type	Accuracy (%)	
			News	Legal
Baseline Model	Manhattan-LSTM	T	70.28	86.06
	tBERT	T	65.51	78.01
	Sentence-BERT (CLS)	T	72.01	86.40
	Sentence-BERT (Avg)	T	72.52	86.66
	Sentence-BERT (Max)	T	72.46	86.84
Proposed Model	BERT (CLS) + LSTM	S	76.57	93.81
	BERT (CLS) + LSTM + LDA	S	**78.54**	**94.20**

The proposed model showed the best performance in evaluating the similarity model used in previous studies in Table 5. It can identify that the document embedding method of all baseline models is a better approach to use global document information than to use local document information through experimental results in a truncate method. However, in the case of tBERT, it showed the worst performance, and an additional experiment was conducted on the use of topic information.

Table 6. Concat strategy results

Concat Strategy	Accuracy (%)	
	News	Legal
u, v	70.28	86.06
$u, v, \lvert u - v \rvert$	65.51	78.01
$u, v, u * v$	72.52	86.66
$u, v, \lvert u - v \rvert, u * v$	**78.54**	**94.20**

When comparing the results of each concat strategy of document embedding, the method of applying the difference between document embedding and dot operation performed the best (Table 6). It can identify that the operation of calculating the difference in embedding has the effect when comparing similarity between documents. The tBERT simply takes a strategy of concatenation each of the two embeddings, so it can be interpreted that the performance degradation occurred in table 5.

Table 7. Inference latencies (seconds) of CLS at document type

Document Type	News	Legal
Truncate (T)	170	356
Segment (S)	728	1,824

In the inference delay time, the experiment was conducted based on the time at which BERT CLS token for each document type occurred. The method of segment documents took 5.69 times more in news data and 5.12 times more in legal data than the method of truncate. It is inevitable to take long information delay seconds because all information is used without omitting the document, but the proposed method in Table 5 can confirm performance improvement by 6.02 and 7.54 values differences in each data. In particular, the length of the document is long, and the domain-specific document is more effective, and the inference latency time is relatively little compared to the news data.

5. Conclusion

In this study, a document embedding and document-like judgement model using document global information was newly proposed by improving the document truncate method used based on the existing BERT in the document embedding method. Documents were divided into segments and then integrated to use all the information in the document, and topical distribution information of segments was also included in the document embedding, built by topic modeling to supplement information within a specific domain document. Especially, it was verified that vector operations using the distance between embeddings after document embedding had a further influence on document-like discrimination. Compared to conventional document embedding methods, document embedding combined with segment topical distribution information showed superior performance not only in common sense fields such as news but also in legal domain data and showed the best results among semantic similarity models.

As future work, the proposed method utilizes all information in the document, resulting in a significant information delay. To reduce time, the method is considered to be needed with a new approach, such as an extension of the maximum possible length applied to BERT or a summary that can sufficiently compress the information in the document.

References

[1] Jones, K. S. (1972). A statistical interpretation of term specificity and its application in retrieval. *Journal of documentation*.

[2] Zhang, X., Zhao, J., & LeCun, Y. (2015). Character-level convolutional networks for text classification. *Advances in neural information processing systems*, *28*, 649-657.

[3] Yoon Kim. (2014). Convolutional Neural Networks for Sentence Classification. *arXiv preprint arXiv:1408.5882.*

[4] Yang, Z., Yang, D., Dyer, C., He, X., Smola, A., & Hovy, E. (2016, June). Hierarchical attention networks for document classification. In *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: human language technologies* (pp. 1480-1489).

[5] Abreu, J., Fred, L., Macêdo, D., & Zanchettin, C. (2019, September). Hierarchical attentional hybrid neural networks for document classification. In *International Conference on Artificial Neural Networks* (pp. 396-402). Springer, Cham.

[6] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. In *Advances in neural information processing systems* (pp. 5998-6008).

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

[8] Adhikari, A., Ram, A., Tang, R., & Lin, J. (2019). Docbert: Bert for document classification. *arXiv preprint arXiv:1904.08398*.

[9] Sun, C., Qiu, X., Xu, Y., & Huang, X. (2019, October). How to fine-tune bert for text classification?. In *China National Conference on Chinese Computational Linguistics* (pp. 194-206). Springer, Cham.

[10] Mueller, J., & Thyagarajan, A. (2016, March). Siamese recurrent architectures for learning sentence similarity. In *Proceedings of the AAAI conference on artificial intelligence* (Vol. 30, No. 1).

[11] Zhu, W., Yao, T., Ni, J., Wei, B., & Lu, Z. (2018). Dependency-based Siamese long short-term memory network for learning sentence representations. *PloS one*, *13*(3), e0193919.

[12] Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent dirichlet allocation. *the Journal of machine Learning research*, *3*, 993-1022.

[13] Ostendorff, M., Bourgonje, P., Berger, M., Moreno-Schneider, J., Rehm, G., & Gipp, B. (2019). Enriching bert with knowledge graph embeddings for document classification. *arXiv preprint arXiv:1909.08402*.

[14] Peinelt, N., Nguyen, D., & Liakata, M. (2020, July). tBERT: Topic models and BERT joining forces for semantic similarity detection. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics* (pp. 7047-7055).

[15] Reimers, N., & Gurevych, I. (2019). Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*.

[16] Joshi, M., Levy, O., Weld, D. S., & Zettlemoyer, L. (2019). BERT for coreference resolution: Baselines and analysis. *arXiv preprint arXiv:1908.09091*.

[17] Wan, L., Papageorgiou, G., Seddon, M., & Bernardoni, M. (2019). Long-length Legal Document Classification. *arXiv preprint arXiv:1912.06905*.

[18] Pappagari, R., Zelasko, P., Villalba, J., Carmiel, Y., & Dehak, N. (2019, December). Hierarchical transformers for long document classification. In *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)* (pp. 838-844). IEEE.

[19] Röder, M., Both, A., & Hinneburg, A. (2015, February). Exploring the space of topic coherence measures. In *Proceedings of the eighth ACM international conference on Web search and data mining* (pp. 399-408).

[20] Chang, J., Gerrish, S., Wang, C., Boyd-Graber, J. L., & Blei, D. M. (2009). Reading tea leaves: How humans interpret topic models. In *Advances in neural information processing systems* (pp. 288-296).

Abstract in Korean

유사 문서 판별을 위한

토픽 모델링 기반의 Siamese-LSTM-BERT 모델

연세대학교 일반대학원

산업공학 전공

김동욱

BERT는 다양한 자연어 처리 작업에서 우수한 성능을 보여주며, 풍부한 표현의 텍스트 임베딩을 생성한다. 그러나 BERT에 적용가능한 텍스트의 길이가 제한되기 때문에 길이가 짧은 문장들에 대해서만 연구가 주로 진행되었다. BERT 기반의 문서 임베딩 생성을 위한 여러 시도가 있었지만, 문서 내용을 발췌하여 문서의 일부분만으로만 임베딩을 생성하였다. 문서 발췌 방법은 문서의 정보 손실을 발생시키기 때문에 정확한 문서 임베딩을 형성하는 데는 한계가 있다. 문서의 발췌 정보를 사용하는 방법은 문서 구조에 대한 경험적인 지식이나 문서 내 중요한 부분을 미리 알아야 하는 조건이 요구된다. 그러나, 특정 도메인에 대한 문서의 경우 전문적인 도메인 지식이 부족할 경우 문서에서 중요한 내용을 알기 어려우며 문서의 유형이 다를 때마다 임베딩 할 문서의 위치를 매번 변경해야 하며, 잘못된 부분을 발췌할 경우 정확하지 않은 문서 임베딩이 생성될 수 있다.

본 연구에서는 BERT를 기반으로 길이가 긴 문서를 위한 문서 임베딩 방법을 제안한다. 연구에서 제안한 모델은 문서를 세그먼트로 나누고 이를 각각의 시퀀스 상태로 간주한다. 이 시퀀스는 LSTM을 통해 하나의 문서 임베딩으로 표현될 수 있으며, 문서의 전체적인 정보를 사용하는 데 도움을 줄 수 있다. 그리고 도메인에 적합한 문서 임베딩 생성을 위해 토픽 모델링을 사용하여 각각의 세그먼트에 대해서 토픽 분포 정보를 결합하여 도메인에 특화된 문서 임베딩을 생성하였다.

제안된 방식의 모델은 Siamese Network를 사용하여 토픽 모델링을 통해 추론된 토픽 분포 정보와 BERT를 통해 추론된 세그먼트가 결합된 문서 임베딩을 기반으로 문서 간의 유사성을 판별하는 작업을 수행한다. 또한, 기존 BERT의 적용가능한 최대 길이 문제를 개선하여 문서의 로컬 정보를 임베딩에 사용하는 대신 문서의 전역 정보를 기반으로 임베딩을 생성할 수 있도록 하고, 토픽 정보를 활용해 도메인에 특화된 문서의 유사성 판별에 기존 연구방법론 보다 향상된 결과를 보여준다.

핵심어 : BERT, 토픽 모델링, Siamese Network, 유사도 판별, 문서 임베딩