An Exploitation of Heterogeneous Graph Neural Network for Extractive Long Document Summarization

Anonymous ACL submission

Abstract

Heterogeneous Graph Neural Networks (HeterGNN) has been recently introduced as an emergent approach for many Natural Language Processing (NLP) tasks by enriching the complex information between word and sentence. In this paper, we try to improve the performance of Extractive Document Summarization (EDS) for long-form documents based on the concept of HeterGNN. Specifically, long documents (e.g., Scientific Papers) are truncated for most neural-based models, which leads to the challenge in terms of information loss of inter-sentence relations. In this regard, we present a new method by exploiting the capabilities of HeterGNN and pre-trained language models. Particularly, BERT is considered for improving the sentence information into the Heterogenous graph layer. Accordingly, two versions of the proposed method are presented which are: i) Multi Graph Neural Network (MTGNN-SUM), by combining both heterogeneous graph layer and graph attention layer; and ii) HeterGNN with BERT (HeterGNN-BERT-SUM), by integrating BERT directly into the heterogeneous graph structure. Experiments on two benchmark datasets of long documents such as PubMed and ArXiv show that our method outperforms state-of-the-art models in this research field.

1 Introduction

Document summarization aims to automatically extract a set of sentences, which represents information for whole document, by ranking the importance of sentence features. Most of previous algorithms require hand-crafted features for sentence representation (Yao et al., 2017). Recently, with the rapid development of Deep Learning (DL) for various Natural Language Processing (NLP) tasks, many DL-based models have been introduced for improving the EDS problem (El-Kassas et al., 2021). Zhang et al. (2016) proposed a simple convolutional neural network (CNN) with pre-trained word embedding for jointly learning and performing sentence features ranking. The experimental results demonstrate the effectiveness of pre-trained word embedding in DL for text summarization comparing with traditional methods.

Notably, GNN, a DL-based approach which operate on graph domain (Zhou et al., 2020a), has introduced as an emergent approach for EDS problem. Specifically, GNN-based models are able to encode the complicated pairwise relationships between entity tokens for better informative representations (Wu et al., 2021). Cui et al. (2020) uses information of topic-aware to change the representation of words to a new representation. Then, a GNN model for capturing relationships efficiently via graph-structured document representation between sentences. Sequentially, recent works focus on HeterGNN, a special kind of GNN (Zhang et al., 2019), for enriching the relationships between words and sentences, which have achieved remarkable results in NLP tasks. Particularly, Wang et al. (2020) presented a heterogeneous graph-based neural network for extractive summarization (HeterSumGraph) by using more fine-grained semantic units in the summarization graph to extract the complex relationships between words and sentences. Accordingly, the model has achieved the top performance in CNN/DailyMail and NYT50 datasets in terms of non-BERT-based approach. In order to utilize the capability of BERT-based models (Devlin et al., 2019), Jia et al. (2020) proposed a hierarchical attentive heterogeneous graph (HAHSum) to improve the redundant phrases problem between extracted sentences of the summarization. HAHSum has achieved remarked results on news article datasets such as CNN/DailyMail, NYT, and Newsroom. However, the model requires external analysis for modeling long-range dependencies.

Observely, since transformer-based language models are not able to process long pieces of texts, there is not much remarkable achievements for

the EDS problem on long documents. Several works have provided promising results (Cui and Hu, 2021), however, the input length limitation and encoding long texts are still open challenges in this research field (Zhong et al., 2020). In this study, we take an investigation on improving the performance of EDS problem for long documents in which the core idea is to exploit the complex relationship of sentences connection. Specifically, based on the advantages of HeterGNN for extracting semantic information between word and sentences, we employ a homogeneous GNN (i.e., Graph Attention Network (GAT)) with BERT for sentence representation to extract the relationship between sentences. In this regard, the proposed combined model is able to capture the semantic information for both inter and intra sentence connections. To the best of our knowledge, this paper is the first study to combine both types of graph structure for the NLP tasks. Specifically, the main contribution of our study is threefold as follows:

- We propose a new approach for learning the complex relationship of sentence connections. Accordingly, two versions of the proposed method are presented for the long document summarization.
- We evaluated the proposed method with two benchmark long documents datasets such as PubMed and ArXiv. The experiential results show that our method outperforms state-of-the-art models for the EDS problem.
- The proposed method is able to extract the complex relationship for both intra and inter sentence relations, which can be easily extended for other NLP tasks (e.g., keyphrases extraction). Our source code is available on [1] for further investigations.

The rest of this paper is organized as follows: Section 2 is a brief review of document extraction and graph neural network models. We present our method with two versions in Section 3. Section 4 reports the evaluation results on two well-known benchmark datasets of long-form documents. The discussions and future works are concluded in Section 5.

[1]Code will be released at https://github.com/

2 Related work

2.1 Extractive Summarization

TextRank (Mihalcea and Tarau, 2004) and LexRank (Erkan and Radev, 2004) was two of traditional methods for extractive summarization. The core idea is to calculate the similar scores between sentences in order to extract the summary sentences. With the rapid growth of DL-based models, neural networks have achieved great success in many NLP tasks, including extractive summarization (Zhang et al., 2018; Dong et al., 2018; Narayan et al., 2018; Cohan et al., 2018; Xiao and Carenini, 2019, 2020).

In recent years, pre-trained language model has become an advanced method in text summarization. Liu and Lapata (2019) proposed a transformer network on BERT representation (BERTSUM) as pretrained encoders to express the semantics of a document. Specifically, the architecture of BERTSUM is illustrated in the Figure 1. Subsequently,

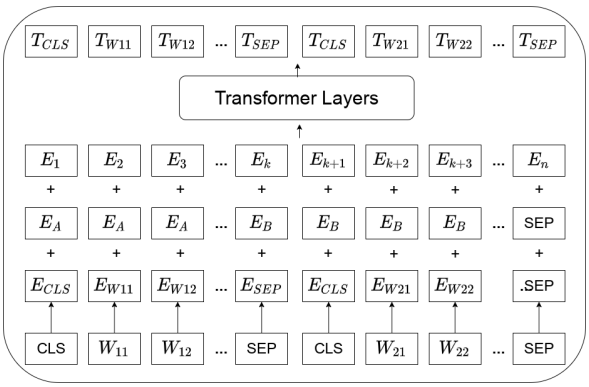

Figure 1: BERTSUM archiecture extends BERT with multiple [CLS] symbols to learn sentence representations and segmentation embeddings.

Zhong et al. (2020) construct Siamese-BERT architecture to match document and candidate summary, which achieve remarked results on CNN/DailyMail dataset. Xu et al. (2020) used discourse information encoded with graph convolution network (GCN) to reduce summarization redundancy and integrate with document encoder by BERT to capture long-range dependencies among discourse units. Yuan et al. (2020) integrated dependency parsing to extract important phrases and present a hierarchical transformer network for improving the performance. Zhou et al. (2020b) proposed an analysis sentence by adopting constituency parsing and using BERT for representing extracted phrases. Then, a transformer network is adopted to extract summary from documents.

2.2 Graph Neural Network

GNN-based models with their variants (e.g., GCN and GAT) have provided the capability for exploiting the sentence relation information encoded in graph representations (Yasunaga et al., 2017; Fernandes et al., 2018). However, the whole graph is assumed to share the same type of nodes, which is not appropriate to exploit the hierarchical problems in many real-word applications. Therefore, Zhang et al. (2019) presents HeterGNN by defining the problem of heterogeneous graph representation learning. Sequentially, HeterGNN-based models have been applied for various downstream applications such as recommendation (Fan et al., 2019) and link prediction (Zheng et al., 2020).

Regarding EDS problem, HeterSumGraph (Wang et al., 2020) is a state-of-the-art model of non-BERT-based summarization. In particular, the model expands the relationship between sentences by introducing word nodes. Figure 2 demonstrates the architecture of HeterSumGraph, which includes three main components such as graph initialization, heterogeneous graph layer, and sentence selection module. However, the complex relationship between sentences, especially the redundant phrases between extracted sentences is not taken into account (Huang and Kurohashi, 2021).

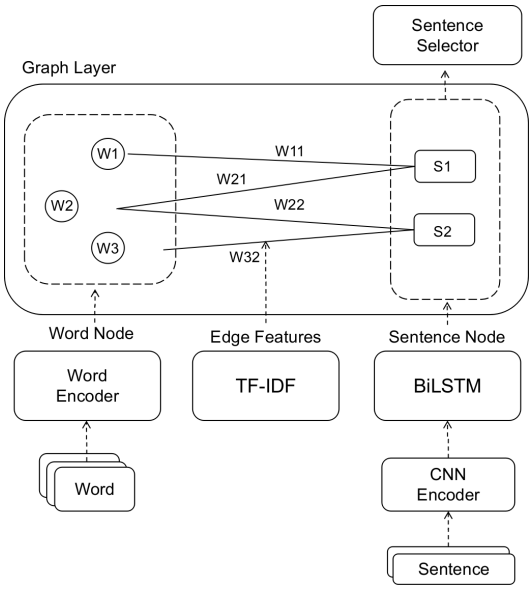

Figure 2: Model overview of HeterGNN model for EDS problem

Therefore, inspired of the work in Wang et al. (2020), this study tries to improve the performance of HeterGNN model by enriching the inter-sentences information for the sentence representation. Specifically, we utilize the capabilities of HeterGNN and BERT for exploiting the complex relationship of sentences connections. The main components of our method are sequentially presented in the following sections.

3 Methodology

Given an arbitrary document $D = \{s_1, .., s_n\}$ consisting n sentences, the objective of EDS problem is to predict a sequence of a set of binary label $\{y_1, .., y_n\}$. Specifically, $y_j \in [0, 1]$ represents the $j - th$ sentence, which should be included in the summary. Our proposed model for EDS problem includes two learning layers, which execute simultaneously, such as heterogeneous graph layer and graph attention layer. More details of our proposed method is described in Section 3.3. Furthermore, a new version by directly integrating BERT into HeterGNN is also taken into account, which is presented in Section 3.4. Specifically, the architecture of Homogeneous and Heterogeneous GNNs are sequentially presented in following Sections.

3.1 Homogeneous Graph Neural Network

Graph Construction: Let $G_1 = \{V_1, E_1\}$ denotes an arbitrary graph, where V_1 and E_1 represent the set of node and edge, respectively. Consequentially, the homogeneous graph an input document can be defined as a set of node $V_1 = s_1, ..., s_n$ where n is the number of sentence in the document.

For the document encoder process, BERT (Devlin et al., 2019) is adopted to generate the local hidden representations between sentences. Specifically, we adopt the concept of BERTSUM(Liu and Lapata, 2019) with multiple CLS for sentence representation. Sequentially, *CLS* and *SEP* tokens are inserted at the beginning and end of each sentence. Then, all tokens are fed into BERT to learn the hidden state, which can be denoted as follows:

$$h_{1,0}, h_{1,1}, ..., h_{n,0}, ..., h_{n,*} = BERT(w_{1,0}, w_{1,1}, ..., w_{n,0}, ..., w_{n,*}) \quad (1)$$

where $w_{i,j}$ represents the *i*-th sentence, and *j*-th word. $w_{i,0}$ and $w_{i,*}$ represents the *CLS* and *SEP* tokens of the *i*-th sentence, $h_{i,j}$ stands for the hidden state of the corresponding token. After BERT encoding, we select the hidden state of *CLS* to represent sentence contextual representations, which is demonstrated as follows:

$$H_B = h_{1,0}, ..., h_{N,0} \quad (2)$$

Sequentially, the document encoder is put into a GAT model for enriching the sentence connections. Figure 3 illustrates the process of the Homogeneous GNN for extraction the sentence-to-sentence relationship. Notably, our method is able to signifi-

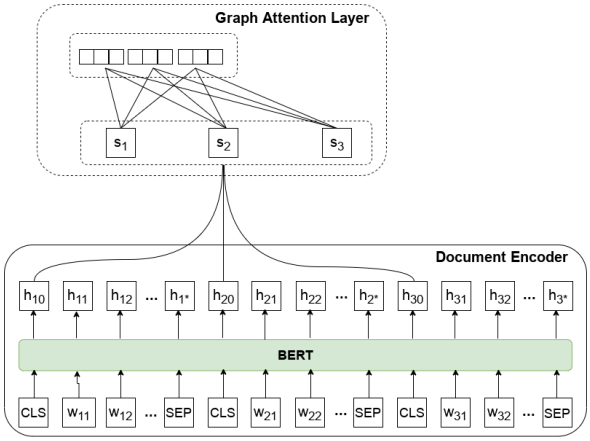

Figure 3: Homogeneous GNN architecture for extracting inter-sentence relations

cantly reduce the computational complexity since we do not need to connect all node sentences as in BERTSUM architecture.

Graph Propagation: Regarding the message passing process, we adopt GAT model (Velickovic et al., 2018) to learn hidden representation of each node by aggregating the information from its neighbors. Specifically, the updated node representation with GAT can be calculated as follows:

$$z_{ij} = LeakyReLU(W_a[W_q h_i; W_e h_j]) \quad (3)$$

where h_i is the *i*-th node representation, σ denotes an activation function, and N_i stand for neighbor nodes. W_a, W_q, W_e, and W_v are trainable weights. Subsequently, the attention score between two sentence node is formulated as follows:

$$\alpha_{ij} = softmax(z_{ij}) = \frac{exp(z_{ij})}{\sum_{l \in N_i} exp(z_{il})}$$
$$\mu_i = \sigma(\sum_{j \in N_i} \alpha_{ij} W_v h_j) \quad (4)$$

Consequentially, the output with multi-head attention can be calculated as follows:

$$h'_i = ||_{k=1}^{K} \sigma(\sum_{j \in N_i} \alpha_{ij}^k W_v^k h_j) \quad (5)$$

where $||*$ represents multi-heads concatenation. Furthermore, a residual connection is adopted to avoid gradient vanishing after iterations. Consequentially, the final output can be updated as follows:

$$H_s^{G_1} = h'_i + h_i \quad (6)$$

In a nutshell, we use GAT for H_B to learn relationship between sentences in document. The output is a representation of sentences, which is concatenated with the output of heterogeneous graph layer for the final representation of a sentence.

3.2 Heterogeneous Graph Neural Network

Graph Construction: Let $G_2 = \{V_2, E_2\}$ denotes an undirected graph for representing the input document. The heterogeneous graph for an input document can be defined as $V_2 = V_w \cup V_s$ and $E_2 = \{e_{11}, ..., e_{mn}\}$, where $V_w = \{w_1, ..., w_m\}$ and $V_s = \{s_1, ..., s_n\}$ represents m unique words and n sentences of a document, respectively. e_{ij} denotes the edge between the *i*-th word and *j*-th sentence. Following the concept of HeterSumGraph (Wang et al., 2020), sentence node features are calculated by combining CNN for extracting the local n-gram feature of each sentence and bidirectional Long Short-Term Memory (BiLSTM) for extracting the sentence-level feature. In this regard, the feature of the sentence s_j can be obtained as follows:

$$X_{s_j} = CNN(x_{1:p}) \oplus BiLSTM(x_{1:p}) \quad (7)$$

where p denotes number of word in the sentence. Furthermore, TFIDF is adopted for further approval information of the relationships between word and sentence, as shown in Figure 1.

Graph Propagation: The heterogeneous graph layer is also updated using GAT, which is defined from Equation 3 to Equation 6. However, the vanilla GAT has designed for homogeneous graphs. Therefore, Wang et al. (2020) has presented a modified GAT and an iterative updating mechanism for heterogeneous graph updated layer. Specifically, the Equation 3 can be re-formulated as follows:

$$z_{ij} = LeakyReLU(W_a[W_q h_i; W_e h_j; \overline{e}_{ij}]) \quad (8)$$

where $\overline{e}_{ij}$ denotes the multi-dimensional embedding space ($\overline{e}_{ij} \in \mathbb{R}^{mn \times d_e}$), which is mapped from edge weight e_{ij}. Sequentially, an iterative updating mechanism is adopted for obtain a new of word node and sentence node. In particular, in order to pass messages between word and sentence nodes, the sentences with their neighbor word nodes are

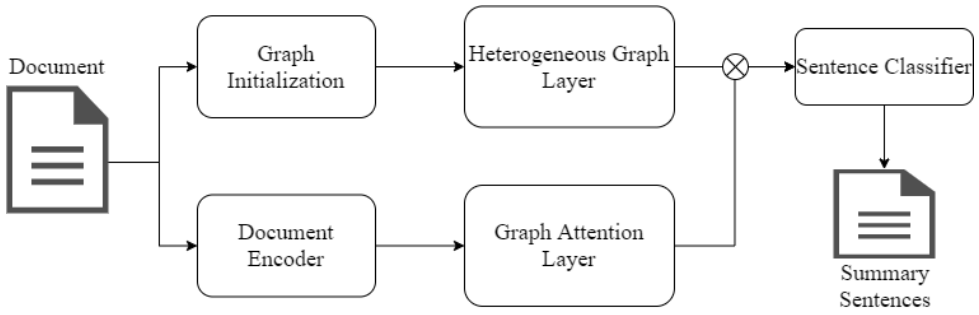

Figure 4: Overview pipeline of the proposed model. Specifically, the model executes simultaneously two phases. In the first phase, the word and sentence nodes were encoded and input to a heterogeneous graph layer (Wang et al., 2020). The other phase encodes the document with pre-trained BERT and inputs in a graph attention layer. The output of two phases is concatenated and put into a MLP layer in order to classify label for each sentence in the document.

updated via modified-GAT and Position-Wise Feed-Forward (FFN) layer, which can be formulated as follows:

$$\begin{aligned} U^1_{s\leftarrow w} &= GAT(H^0_s, H^0_w, H^0_w) \\ H^1_s &= FFN(U^1_{s\leftarrow w} + H^0_s) \end{aligned} \tag{9}$$

where H^0_w (H^1_w) and H^0_s are the node features of word X_w ($X_w \in \mathbb{R}^{m\times d_w}$) and sentence X_s ($X_s \in \mathbb{R}^{n\times d_s}$), respectively. Note that H^0_s is used as the attention query and H^0_w are regarded as key and value. Sequentially, the new representations of word node can be obtained using the updated sentence nodes and further updated sentence or query nodes, iteratively. Specifically, each iteration contains a sentence-to-word and a word-to-sentence update process, which is formulated as follows:

$$\begin{aligned} U^{t+1}_{w\leftarrow s} &= GAT(H^t_w, H^t_s, H^t_s) \\ H^{t+1}_w &= FFN(U^{t+1}_{w\leftarrow s} + H^t_w) \\ U^{t+1}_{s\leftarrow w} &= GAT(H^t_s, H^{t+1}_w, H^{t+1}_w) \\ H^{t+1}_s &= FFN(U^{t+1}_{s\leftarrow w} + H^t_s) \end{aligned} \tag{10}$$

3.3 Multi Graph Neural Network for EDS

Figure 4 illustrates the pipeline of our multi GNN models. Specifically, the outputs of sentence features from two aforementioned layers are then concatenated for the final representation, which is formulated as follows:

$$H = H^{Homo}_s \oplus H^{Heter}_s \tag{11}$$

Observably, by concatenating the outputs of two aforementioned graph layers, final representation includes the information of both intra and inter-sentence relations. Sequentially, the output of the concatenation is put into a sentence classier for ranking the classification.

3.4 Hetergogeneous GNN with BERT

As mentioned above, we consider another version of the proposed method by integrating sentence representations from BERT into Heterogeneous GNN. Accordingly, the selected hidden state of CLS are integrated for extract sentence features. The architecture of the integrated Hetergogeneous GNN with BERT is illustrated in Figure 5. In this regard, the feature of a sentence (Equation 12) can be re-formulated as follows:

$$X_{s_j} = CNN(x_{1:p}) \oplus BiLSTM(x_{1:p}) \oplus H_B(s_j) \tag{12}$$

Sequentially, the new feature sentence is input into HeterGNN model for leaning the complex relationship between word and sentences.

3.5 Sentence Classifier

We execute node classification method for sentences, which are ranked by the scores. Sequentially, cross-entropy loss is used for classifying sentences, which is formulated as follows:

$$\mathcal{L} = \sum_{i=1}^{n} y_i log(\hat{y}_i) + (1 - y_i) log(1 - \hat{y}_i) \tag{13}$$

4 Experiments

4.1 Experimental Setup

Dataset: Extracting summarization of news articles has been widely explored during recent years, however, longer documents are still challenge issues due to the accurately encoding problem of long texts for the summarization. In this regard, we focus on evaluating the proposed method with various length of the documents. Specifically, two long document datasets are taken into account for the

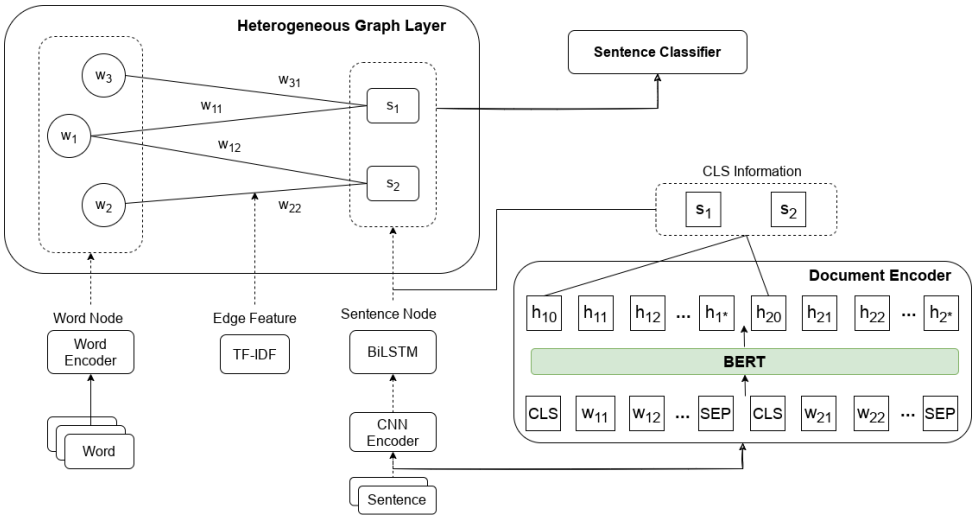

Figure 5: Overview of the integrated Heterogeneous GNN model with BERT.

evaluation. The statistics of benchmark datasets are illustrated in Table 1. Accordingly, PubMed[2] and arXiv[3] are standard datasets for long documents, which are scientific papers. For the data processing, we use the same split as the work in Cohan et al. (2018) to process arXiv and PubMed dataset for the evaluation and follow Liu and Lapata (2019) to get ground-truth labels.

	Dataset	PubMed	arXiv
Docs	Train	119,924	203,037
	Val	6,633	6,436
	Test	6,658	6,440
Tokens	Doc.	3,016	4,938
	Sum.	203	220

Table 1: Statistics of evaluated datasets.

Evaluated Models: We evaluate our method on two well-known long document datasets (i.e., scientific papers) and compare with previous state-of-the-art EDS models, which are classified into different approaches such as approaches without pre-trained language models, BERT-based models, and Graph-based models. Specifically, results of evaluated models are obtained from respective papers. More detail of those aforementioned evaluated models are presented in the following section. Notably, with referring as a part of our model, we re-execute the HeterSumGraph model following the guidelines of the original paper [4]. Furthermore, each proposed model is executed three-time and calculated mean values for final reports.

Hyperparameter Setting: Regarding the encoding, the vocabulary is limited to 50,000 and the tokens are initialized with 300-dimensional with Glove embedding. The dimension of sentence node and edge features are set to 128 and 50, respectively. The number of multi-head in each GAT layer is set to 8. For document encoder, we use bert-base-uncased version of BERT and fine-tune for the experiments. In case of decoding process, we select top-6 and top-5 for PubMed and arXiv datasets, respectively, according to the best performance of validation set. The maximum number of sentences in each document is set to 150, which is suitable with our limited computational resource. More analysis of the length of sentences are presented in the following section. The model is trained with Adam optimizer. The learning rate is set to 1e-3 and use early stop with each three epochs. Moreover, learning rate decay is used after each epoch to improve the performance. All models are trained on a single Tesla V100 32GB GPU, which have completed the training process with around 10 epochs. The total time for each epoch with the best model is around 6 hours and 3 hours for PubMed and arXiv datasets, respectively. Note that, since we focus on long documents, the computational time is quite high. Therefore, we do not use hyperparameter optimization for improving the performance.

4.2 Experimental Results

The comparison models are divided into different parts. The first part reports the Lead-3 and Ora-

[2]https://pubmed.ncbi.nlm.nih.gov/
[3]https://arxiv.org/
[4]Source: https://github.com/brxx122/HeterSumGraph

Model	PubMed			arXiv		
	R-1	R-2	R-L	R-1	R-2	R-L
SumBasic*	37.15	11.36	33.43	29.47	6.95	26.30
LexRank*	39.19	13.89	34.59	33.85	10.73	28.99
LSA*	33.89	9.93	29.70	29.91	7.42	25.67
Oracle (Xiao and Carenini, 2020)	55.05	27.48	49.11	53.89	23.07	46.54
SummaRuNNer+	43.89	18.78	30.36	42.91	16.65	28.53
Seq2seq-attentive+	44.81	19.74	31.48	43.58	17.37	29.30
Seq2seq-cancat+	44.85	19.70	31.43	43.62	17.36	29.14
Cheng&Lapata (2016)+	43.89	18.53	30.17	42.24	15.97	27.88
Attn-Seq2Seq*	31.55	8.52	27.38	29.30	6.00	25.56
Pntr-Gen-Seq2Seq*	35.86	10.22	29.69	32.06	9.04	25.16
Discourse-aware*	38.93	15.37	35.21	35.80	11.05	31.80
ExtSum-LG (Xiao and Carenini, 2020)	45.39	20.37	40.99	44.01	17.79	39.09
MATCHSUM (Zhong et al., 2020)	41.21	19.41	36.75	40.59	12.98	32.64
Topic-graphSum (Cui and Hu, 2021)	45.95	20.81	33.97	44.03	18.52	32.41
SSN-DM (Cui and Hu, 2021)	46.73	21.00	34.10	45.03	**19.03**	32.58
MTGNN-SUM	**48.42**	**22.26**	**43.66**	46.39	18.58	40.50
HeterGNN-BERT-SUM	47.85	21.64	43.13	**46.52**	18.62	**40.68**

Table 2: Results on PubMed and arXiv datasets. Report results with * are from Cohan et al. (2018), and results with + are from Xiao and Carenini (2019). Other results are obtained from respective papers.

cle. The second part shows results of the approach without pre-trained language models. The third approach includes BERT-based models. The next section presents the result of graph-based approach including the models with document-level approach, which requires different levels of information such as words, sentences, topic, and spotlights redundancy dependencies between sentences. The last section is our proposed models, which include two versions such as the multi GNN (MTGNN-SUM) and the integrated Heterogeneous GNN model with BERT representation (HeterGNN-BERT-SUM).

Table 2 shows the results of our method comparing with state-of-the-art models on PubMed and arXiv, respectively. Accordingly, our results are mostly outperforms state-of-the-art models in this research field. In particular, only R-2 of SSN-DM, the lasted state-of-the-art model is slightly better than our method in case of arXiv datasets. However, the R-L metric of our method is significantly improved comparing with SSN-DM model. Specifically, our method is significant effective with Pubmed datasets using MTGNN-SUM model. Meanwhile, HeterGNN-BERT-SUM are slight better than MTGNN-SUM in terms of arXiv dataset. This result indicates the important of exploiting the relationship between sentences for improve the performance of long document summarization. Furthermore, the issue of data dependence may require different configurations. We leave this issue in other study regarding this study.

4.3 Quality Analysis

Ablation Study. In our model, we enrich the complex relationships by exploring both heterogeneous graph and homogeneous graph operation for the sentence connection. In order to explore the effect of each component, we design different variants of our method as follows:

- **HomoGraph-SUM**: only uses the graph attention layer for document encoding to extract inter-sentence relationships. The model is designed following the description in Section 3.1 of Homogeneous Graph Neural Network.
- **HeterGraph-SUM**: only use heterogeneous graph layer which contains semantic nodes to enrich the cross-sentence relations. Specifically, HeterGrap-Sum is designed following the description in Section 3.2.

The results of those aforementioned variants of our model on benchmark datasets are presented tin Tab. 3. Accordingly, using only GAT layer with BERT encoder gets worse results. Furthermore, integrating document encoder inside the heterogeneous graph is not better than only using only using

Dataset	Model	R-1	R-2	R-L
PubMed	HomoGraph-SUM	39.29	13.74	34.49
	HeterGraph-SUM	46.03	19.79	41.48
	MTGNN-SUM	48.42	22.26	43.66
arXiv	HomoGraph-SUM	41.13	13.11	35.84
	HeterGraph-SUM	45.06	16.97	39.38
	MTGNN-SUM	46.39	18.58	40.50

Table 3: Ablation study on benchmark datasets.

heterogeneous graph layer. Consequentially, executing message passing across sentences in our proposed model by combining both graph structures operation is able to to achieve better results.

Length of Document. In this study, we set the maximum number of sentences in each document equals 150 due to our limited computational resources. Though, we are able to improve the performance by learning whole length sentences of the datasets, which include many documents with more than 200 sentences. In order to evaluate the importance of the document length value, we tested our model with the maximum number of sentences are 50 and 100 sentences, respectively. The results of the test models on different values of maximum document sizes are shown in Table 4. Accordingly,

Dataset	Model	R-1	R-2	R-L
PubMed	MTGNN-SUM-50	46.20	20.04	41.58
	MTGNN-SUM-100	47.85	21.64	43.13
	MTGNN-SUM-150	48.42	22.26	43.66
arXiv	MTGNN-SUM-50	44.91	16.89	39.14
	MTGNN-SUM-100	46.09	17.98	40.29
	MTGNN-SUM-150	46.39	18.58	40.50

Table 4: Results of proposed model with different length of sentences on benchmark datasets.

by increasing the maximum length of sentences, the performances are significantly improved. Consequentially, the results indicated that tuning max length of sentence value is able to enhance the performance. Specifically, we take this issue into account for the future work of this study by executing our model with longer maximum size of documents.

Case Study Figure 6 demonstrates an example of a document with 40 sentences. Accordingly, our model is able to extract the sentences for the in all the positions of the whole document, which indicates the advantage of our method for capturing the long-form texts.

Document:
on february 13 , 1996 , a 7-year - old boy from doihue in administrative region vi was admitted to the hospital clnico fusat of rancagua in the region (figure 1) with a 2-day history of adynamia and dizziness . ... *these analyses identified a rabies antigenic variant associated with tadarida brasiliensis (free - tailed bat) in chile , which had been designated as antigenic variant 4 (agv4) (9,17) . genetic characterization was done by sequencing a 320-bp portion of the rabies virus nucleoprotein gene from nucleotide position 1,157 to 1,476 , as compared with the sadb 19 strain (18,19)* . briefly , genomic viral rna was extracted from infected tissue by using trizol (invitrogen , san diego , ca , formerly gibco - brl inc .) according to the manufacturer s instructions ... phylogenetic analyses of the chilean human isolate demonstrated that it segregated in group d. this group represents the genetic variant of rabies virus most frequently isolated throughout the country , formed by viruses from the metropolitan region and regions iv , v , vi , vii , viii , ix , and x (figure 1) . ... *the absence of a history of an animal bite , the clinical presentation of the disease without the classic signs of hydrophobia or aerophobia , and the absence of any human rabies cases for a period of 24 years in chile were the primary reasons that rabies was not first suspected and a definitive diagnosis was delayed in this case* . retrospective studies of human rabies epidemiology have demonstrated that it is not uncommon to observe rabies cases in which there is no history of a bite , mainly in situations involving insectivorous bat rabies variants . ... finally , there may not be an opportunity to obtain a history from a pediatric patient or to discern an exposure that occurs during sleep or other circumstances (24) . in cases in which a patient shows clinical signs of central nervous system involvement of unknown or suspected viral origin. health - care providers should be aware of the importance of conducting a thorough medical history to appropriately assess the possibility of rabies . with the important changes in the epidemiologic patterns of rabies in latin america , this disease should be included in the differential diagnosis of neurologic diseases characterized by acute encephalitis and progressive paralysis , even when no previous history of an animal bite exists and even in regions where canine rabies has been eradicated .

Reference:
the first human rabies case in chile since 1972 occurred in march 1996 in a patient without history of known exposure .antigenic and genetic characterization of the rabies isolate indicated that its reservoir was the insectivorous bat tadarida brasiliensis . this is the first human rabies case caused by an insectivorous bat rabies virus variant reported in latin america

Figure 6: An example document and gold summary in the PubMed dataset. The words in italics refer to the sentences selected by the greedy algorithm and the underlines sentences are our model-selected summary.

5 Conclusion

This paper presents a novel graph-based method for EDS problem which focuses on exploiting the complex relationship for both inter and intra sentence connection of the long documents. Specifically, long documents mostly are truncated by using neural models, which is the cause of loss information, especially for extractive models. Therefore, we take pretrained models (i.e., BERT) into account for generating the local hidden representations between sentences and put into heterogeneous graph layer for learning the complex relationship of sentences connections. Specifically, two versions of the proposed method are presented and evaluate on two benchmark datasets of long documents (e.g., PubMed and arXiv). The experiments on two well-known long document datasets show promising results of our method.

References

Arman Cohan, Franck Dernoncourt, Doo Soon Kim, Trung Bui, Seokhwan Kim, Walter Chang, and Nazli Goharian. 2018. A discourse-aware attention model for abstractive summarization of long documents. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 615–621, New Orleans, Louisiana. Association for Computational Linguistics.

Peng Cui and Le Hu. 2021. Sliding selector network with dynamic memory for extractive summarization of long documents. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021, Online, June 6-11, 2021*, pages 5881–5891. Association for Computational Linguistics.

Peng Cui, Le Hu, and Yuanchao Liu. 2020. Enhancing extractive text summarization with topic-aware graph neural networks. In *Proceedings of the 28th International Conference on Computational Linguistics, COLING 2020, Barcelona, Spain (Online), December 8-13, 2020*, pages 5360–5371. International Committee on Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.

Yue Dong, Yikang Shen, Eric Crawford, Herke van Hoof, and Jackie Chi Kit Cheung. 2018. BanditSum: Extractive summarization as a contextual bandit. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3739–3748, Brussels, Belgium. Association for Computational Linguistics.

Wafaa S. El-Kassas, Cherif R. Salama, Ahmed A. Rafea, and Hoda K. Mohamed. 2021. Automatic text summarization: A comprehensive survey. *Expert Syst. Appl.*, 165:113679.

Günes Erkan and Dragomir R. Radev. 2004. Lexrank: Graph-based lexical centrality as salience in text summarization. *J. Artif. Int. Res.*, 22(1):457–479.

Shaohua Fan, Junxiong Zhu, Xiaotian Han, Chuan Shi, Linmei Hu, Biyu Ma, and Yongliang Li. 2019. Metapath-guided heterogeneous graph neural network for intent recommendation. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2019, Anchorage, AK, USA, August 4-8, 2019*, pages 2478–2486. ACM.

Patrick Fernandes, Miltiadis Allamanis, and Marc Brockschmidt. 2018. Structured neural summarization. *CoRR*, abs/1811.01824.

Yin Jou Huang and Sadao Kurohashi. 2021. Extractive summarization considering discourse and coreference relations based on heterogeneous graph. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, EACL 2021, Online, April 19 - 23, 2021*, pages 3046–3052. Association for Computational Linguistics.

Ruipeng Jia, Yanan Cao, Hengzhu Tang, Fang Fang, Cong Cao, and Shi Wang. 2020. Neural extractive summarization with hierarchical attentive heterogeneous graph network. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3622–3631, Online. Association for Computational Linguistics.

Yang Liu and Mirella Lapata. 2019. Text summarization with pretrained encoders. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3730–3740, Hong Kong, China. Association for Computational Linguistics.

Rada Mihalcea and Paul Tarau. 2004. TextRank: Bringing order into text. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, pages 404–411, Barcelona, Spain. Association for Computational Linguistics.

Shashi Narayan, Shay B. Cohen, and Mirella Lapata. 2018. Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1797–1807, Brussels, Belgium. Association for Computational Linguistics.

Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2018. Graph attention networks. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.

Danqing Wang, Pengfei Liu, Yining Zheng, Xipeng Qiu, and Xuanjing Huang. 2020. Heterogeneous graph neural networks for extractive document summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6209–6219, Online. Association for Computational Linguistics.

Lingfei Wu, Yu Chen, Kai Shen, Xiaojie Guo, Hanning Gao, Shucheng Li, Jian Pei, and Bo Long. 2021. Graph neural networks for natural language processing: A survey. *CoRR*, abs/2106.06090.

Wen Xiao and Giuseppe Carenini. 2019. Extractive summarization of long documents by combining

global and local context. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3011–3021, Hong Kong, China. Association for Computational Linguistics.

Wen Xiao and Giuseppe Carenini. 2020. Systematically exploring redundancy reduction in summarizing long documents. In *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing*, pages 516–528, Suzhou, China. Association for Computational Linguistics.

Jiacheng Xu, Zhe Gan, Yu Cheng, and Jingjing Liu. 2020. Discourse-aware neural extractive text summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics.

Jin-ge Yao, Xiaojun Wan, and Jianguo Xiao. 2017. Recent advances in document summarization. *Knowl. Inf. Syst.*, 53(2):297–336.

Michihiro Yasunaga, Rui Zhang, Kshitijh Meelu, Ayush Pareek, Krishnan Srinivasan, and Dragomir Radev. 2017. Graph-based neural multi-document summarization. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pages 452–462, Vancouver, Canada. Association for Computational Linguistics.

Ruifeng Yuan, Zili Wang, and Wenjie Li. 2020. Fact-level extractive summarization with hierarchical graph mask on BERT. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 5629–5639, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Chuxu Zhang, Dongjin Song, Chao Huang, Ananthram Swami, and Nitesh V. Chawla. 2019. Heterogeneous graph neural network. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2019, Anchorage, AK, USA, August 4-8, 2019*, pages 793–803. ACM.

Xingxing Zhang, Mirella Lapata, Furu Wei, and Ming Zhou. 2018. Neural latent extractive document summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 779–784, Brussels, Belgium. Association for Computational Linguistics.

Yong Zhang, Meng Joo Er, and Mahardhika Pratama. 2016. Extractive document summarization based on convolutional neural networks. In *IECON 2016 - 42nd Annual Conference of the IEEE Industrial Electronics Society, Florence, Italy, October 23-26, 2016*, pages 918–922. IEEE.

Jianming Zheng, Fei Cai, Yanxiang Ling, and Honghui Chen. 2020. Heterogeneous graph neural networks to predict what happen next. In *Proceedings of the 28th International Conference on Computational Linguistics, COLING 2020, Barcelona, Spain (Online), December 8-13, 2020*, pages 328–338. International Committee on Computational Linguistics.

Ming Zhong, Pengfei Liu, Yiran Chen, Danqing Wang, Xipeng Qiu, and Xuanjing Huang. 2020. Extractive summarization as text matching. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6197–6208, Online. Association for Computational Linguistics.

Jie Zhou, Ganqu Cui, Shengding Hu, Zhengyan Zhang, Cheng Yang, Zhiyuan Liu, Lifeng Wang, Changcheng Li, and Maosong Sun. 2020a. Graph neural networks: A review of methods and applications. *AI Open*, 1:57–81.

Qingyu Zhou, Furu Wei, and Ming Zhou. 2020b. At which level should we extract? an empirical analysis on extractive document summarization. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 5617–5628, Barcelona, Spain (Online). International Committee on Computational Linguistics.