SGAN4AbSum: A Semantic-Enhanced Generative Adversarial Network for Abstractive Text Summarization

Tham Vo (thamvth@tdmu.edu.vn)
Thu Dau Mot University https://orcid.org/0000-0001-7291-4168

Research Article

Keywords: GAN, BERT, GCN, abstractive summarization.

Posted Date: July 30th, 2021

DOI: https://doi.org/10.21203/rs.3.rs-648146/v1

SGAN4AbSum: A Semantic-Enhanced Generative Adversarial Network for Abstractive Text Summarization

Tham Vo[1, *]

[1]*Thu Dau Mot University, Binh Duong, Vietnam*

[1]*thamvth@tdmu.edu.vn*

(*) *Corresponding author*

ABSTRACT

In abstractive summarization task, most of proposed models adopt the deep recurrent neural network (RNN)-based encoder-decoder architecture to learn and generate meaningful summary for a given input document. However, most of recent RNN-based models always suffer the challenges related to the involvement of much capturing high-frequency/reparative phrases in long documents during the training process which leads to the outcome of trivial and generic summaries are generated. Moreover, the lack of thorough analysis on the sequential and long-range dependency relationships between words within different contexts while learning the textual representation also make the generated summaries unnatural and incoherent. To deal with these challenges, in this paper we proposed a novel semantic-enhanced generative adversarial network (GAN)-based approach for abstractive text summarization task, called as: SGAN4AbSum. We use an adversarial training strategy for our text summarization model in which train the generator and discriminator to simultaneously handle the summary generation and distinguishing the generated summary with the ground-truth one. The input of generator is the jointed rich-semantic and global structural latent representations of training documents which are achieved by applying a combined BERT and graph convolutional network (GCN) textual embedding mechanism. Extensive experiments in benchmark datasets demonstrate the effectiveness of our proposed SGAN4AbSum which achieve the competitive ROUGE-based scores in comparing with state-of-the-art abstractive text summarization baselines.

Keywords: GAN; BERT; GCN; abstractive summarization;

1. INTRODUCTION

Along with the tremendous growth of Internet, people are recent more overwhelmed with the dramatic amount of online contents/documents from multiple large-scale digital online resources, e.g. online news platform, social networks, etc. Thus, it is necessary to build up system which support to provide short descriptive contents for full-length documents while still conveying important information in the original source texts. This task is normally known as automatic text summarization [1] [2] which enable users to easy obtain the key information and overall meaning of a given document without reading all of its content. Considering as a primitive application in natural language processing (NLP) area, the text summarization (either extractive or abstractive) is the process of automatically generating summaries (in form of short-length descriptive contents/abstracts) from a natural language document. A qualitied text summarization system should be capable to generate meaningful summaries with important and salient information for length-varied input documents. In general, text summarization can be categorized into extractive and abstractive approach. The extractive text summarization based models tend to learn the important phrases/textual sections of the training source texts to characterize the salient pattern features which are later used to produce the fluent summaries. In extractive summarization approach, the generated summaries are normally composed by a set of phrases/sentences from the original source texts. In more details, the extractive summarization based models produce summaries by selecting a subset of important phrases/sentences from the original texts within the control of predefined compression rate over the length

of generated summaries. In contrast to extractive summarization approach which mainly select informative phrases/sentences in the input documents, the abstractive summarization models are designed to aim at effectively generating new/shorter textual contents which fully reflect critical and salient information of the given source texts. An abstractive summarization technique is only considered as effective if it is capable to produce natural and linguistically coherent summaries which cover principal information of input documents. In order to do this, the proposed abstractive model must be integrated with complex natural language processing mechanisms in order to sufficiently understand the interpret the original document into a shorter and rich-informative form. Therefore, abstractive text summarization task is normally considered as more challenging than the extractive one.

1.1. Recent progress & existing challenges

With the dramatic progresses of deep learning, RNN-based sequence-to-sequence (seq2seq) [3] [4] based neural architecture has become the mainstream for most of recent advanced techniques [5] [6] [7] [8] [9] in text summarization task. Such as the proposals of well-known RNN-based linguistic architecture of Rush, A. M. et al. [5] in applying complex sequential neural architectures to encode/decode deep latent feature representations of sentences for effectively handling abstractive sentence summarization task. The demonstration of significant improvements in accuracy performance have proven the usefulness of the utilization of deep sequential encoder-decoder architecture in text summarization domain. However, the early RNN-based text summarization model still suffered key problems related to the unnatural/influent representation and repeated phrases in generated summaries due to the lack of thorough evaluations on the contextual meanings and semantic relationships between words. Moreover, RNN-based models also encounter the out-of-vocabulary (OOV) regarding challenge in which the system is trained with a fixed set of input and output vocabularies. Thus, it prevents the proposed model to generate meaningful summaries with the representations of new words.

Go along with the first introduction of attention-based mechanism and deep neural transformer architecture [10] [11] in the traditional seq2seq approach, there novel pre-trained textual embedding based models have been proposed, e.g., "*extreme summarization*" (ES) [12], "*multi-news*" (MNSum) [13], "*discourse-aware*" DASum [14], etc. These attention-based sequential neural architectures have shown remarkable improvement in accuracy performance in abstractive summarization task. However, since the abstractive summarization models are designed to concentrate on the process of textual latent feature learning and interpreting the documents into few-sentence summaries, attention-based sequential neural approach also encounter major challenges related to the capability of focusing on salient information in specific sections of documents as well as representing generated summaries with out-of-vocabulary words. There are recent attempts [15] [16] [17] on the application of pre-trained masked linguistic embedding model to overcome the OOV-related and influent summary representation challenges. These pre-trained masked linguistic model have successfully archive state-of-the-art performance in abstractive text summarization task by utilizing available multi-tasked rich-contextual pre-trained linguistic embedding mechanism. The pre-trained linguistic representation learning models (e.g., BERT [10]) has been trained with large-scale text corpora, thus they can sufficiently cover most of contextual information of given input documents to deeply understand and generate meaningful summaries. However, these recent pre-trained based abstractive summarization models are hindered by the limitation of the seq2seq neural architecture in generating trivial and generalized summaries, often involving in common linguistic writing styles and high-frequency phrase occurrence of the training set. Moreover, these sequential pre-trained based summarization techniques also encounter problems related to latent feature ambiguity and inaccurate textual encoding of long-length texts due to the lack of capability in capturing the long-range dependent relationships between words in long documents. To deal with challenges related to the context-varied and long document latent feature representation learning for abstractive summaries, there are integrated GAN [18] with reinforcement learning (RL) based approaches [19] [20] [21] [22]. Recently, adversarial network has become an important downstream learning baseline for multiple domains including NLP which enable to produce expected outputs in forms of real-fake differentiable data generation/validation upon the multi-tasked training

objectives. The GAN [18] is also recently applied in text summarization task with the designed goal of generator the is to produce the summaries for corresponding input documents. The training objective of the generator is to generate summaries which are look like the human-written one in order to fool the discriminator. For the generator, most of recent models [19] [20] utilize the reinforcement learning [23] training strategy to optimize the highly rewarded summaries which fool the discriminator much. However, these GAN-RL based methods also lack of thorough analysis on the sequential and long-range dependent relationships between words in the given training source texts which might lead to the downgrade in quality of the generated summaries. Moreover, recent GAN-based summarization techniques [19] [20] also lack of evaluation on the aspects of context diversity and complex sequential relationships between words in the input documents which might lead to the generation of unnatural and influent summaries in the after all.

1.2. Our contributions in this paper

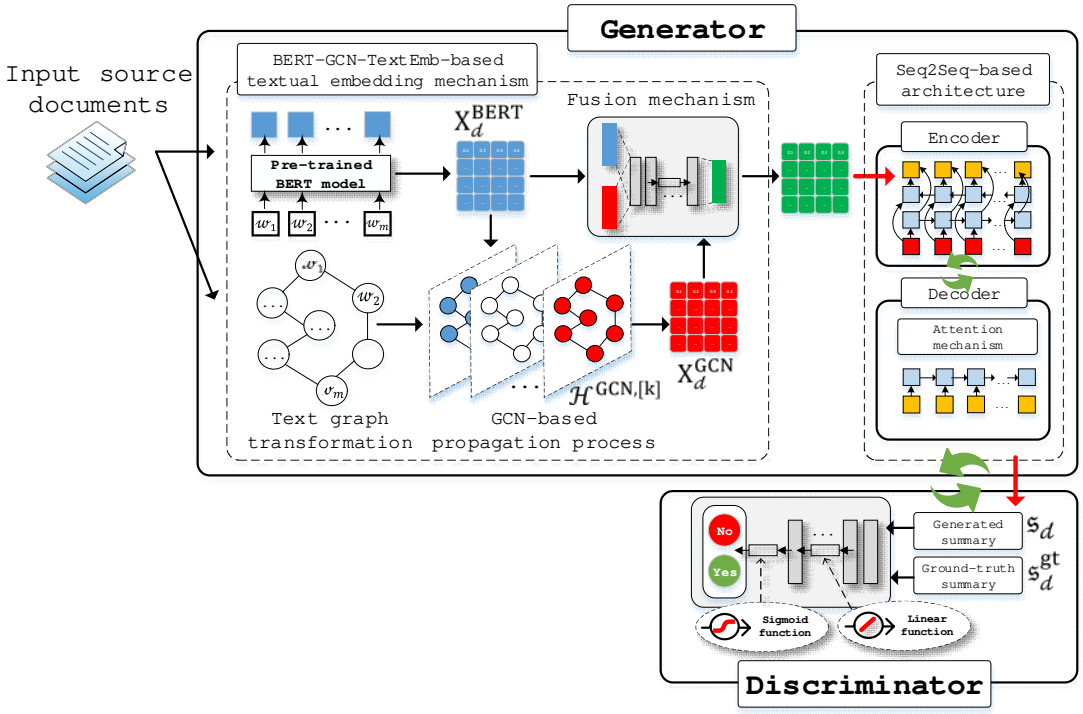

Figure 1. Illustration of the overall architecture of our proposed SGAN4AbSum model

To meet above challenges, in this paper we proposed a novel semantic-enhanced abstractive text summarization model with the integration of GAN and reinforcement learning strategy, called as SGAN4AbSum (as shown in Figure 1). First of all, in order to jointly capture the rich-semantic sequential and global long-range dependent relations between words in given training texts, we apply a combined pre-trained BERT and graph convolutional network [24] (GCN)-based textual representation learning mechanism, called as BERT-GCN-TextEmb method. For the use of GCN as textual encoder, we firstly present the original source texts as n-hop co-occurring relation-based document's graphs, then a multi-layered GCN architecture is applied to learn the long-range dependent structural latent representations of all words in each document's graph. Then, the GCN-based textual embedding matrices are merged into the BERT-based textual representations to form the unified embedding vector of words. Similar to the original architecture of GAN [18], we also have the separated generator and discriminator components as the backbone of our proposed SGAN4AbSum model. For the generator, it receives the raw input documents as the initial inputs, then documents are passed through the BERT-GCN-TextEmb to capture the joint sequential and structural latent features of words in form of embedding vectors. Then, an attention-based neural encoder-decoder architecture is applied to aggregate the input word representations to generate the

associated summaries. For the discriminator component, it is designed to distinguish the golden/ground-truth summaries (real) from the generated ones (fake) which are automatically produced by the generator. To sum up, our contribution in this paper can be summarized as threefold, which are:

- First of all, we introduce an integrated textual embedding method, namely BERT-GCN-TextEmb which enables to jointly learn the rich-contextual sequential and long-range dependent relationships of texts to facilitate the process of summary generation task. By using the pre-trained BERT textual encoder, we can sufficiently capture the representations of all words in a document within different contexts which have been involved in the language-specific pre-trained BERT model. Then, the multi-layered GCN architecture is applied to effectively learn the n-hop co-occurring relationships between words within a given document in form of the text-graph structure. Thus, the combined sequential embedding vectors of words which are produced by BERT-GCN-TextEmb method convey both rich-contextual sequential and long-range dependent latent features of texts.
- Next, the BERT-GCN-TextEmb method is utilized in the generator to learn the rich-semantic representations of words from the input texts. Then, learnt word embedding vectors are fed into an attention-based encoder-decoder architecture to fulfill the abstractive text summarization task. Taking BERT-GCN-TextEmb-based word embedding vectors as the input, the encoder (as a Bi-LSTM architecture) encodes them and generates corresponding hidden states, which are then combined with the context/state vectors of decoder (as an attention-based LSTM architecture) to produce summaries as the sequence of predicted words with a full-connected layer and softmax classification function at the end. To optimize corresponding parameters, we model our generator as the stochastic policy gradient [25] [26] of reinforcement learning approach in GAN.
- Finally, to demonstrate the effectiveness of our proposed SGAN4AbSum we conducted extensive experiments in CNN/Daily-Mail benchmark dataset to demonstrate the effectiveness of our proposed model in comparing with recent state-of-the-art abstractive text summarization baselines. Experimental outputs in terms of ROUGE-based metric prove the usefulness of our proposed ideas in this paper.

The left contents of our paper are organized into 4 sections. In the next section, we briefly provide recent studies in abstractive text summarization as well as discuss about achievements and existing challenges. In the third section, we formally present the methodology and detailed implementations of our proposed SGAN4AbSum model. Next in the fourth section, we present extensive experiments and discussions about the outputs. In this experimental section, we also provide extra ablation studies related to proposed model's parameters. Finally, we conclude about our achievements with the proposed SGAN4AbSum model and highlight some possible improvements for our future works in the fifth section.

2. RELATED WORKS

Thanks to the dramatic increases of large-scale online digital contents on the Internet, people as well as organizations need the supports of text summarization as an indispensable application which enables to effectively capture important information from the given full-length source texts without reading all their inside contents. With the tremendous emerge of deep learning, multiple complex sequential neural architectures have been widely applied to efficiently encode textual information into latent embedding spaces for facilitating the text summarization task. Such as an early well-known works of Rush, A. M. et al. [5] in proposing an attention-based dual RNN-based architecture to encode the input documents into fixed latent embedding vectors then using these document representations to generate new output sequences as summaries from the given source texts by using another RNN architecture. Similar to that, next seq2seq-based text summarization models [6] [7] [9] have performed remarkable improvements in abstractive summarization task. Along with recent advances in NLP domain, the text summarization model is equipped with the powerful attention-based textual learning mechanism, such as the famous "*get-to-the-point*" (GTTP) model [9] of See, A. et al. which proposed a hybrid pointer-generator based approach for effectively handling attention-based abstractive generation process. Similar to that, Merity, S. et al. [27] proposed a

combination of applying pointer-based embedding mechanism to softly matching the original word representation learning layers into the corresponding contextual decoding layers to efficiently generate meaningful abstractive summaries. However, attention-driven RNN-based models also suffer challenges related to the out-of-vocabulary (OOV) representation and incoherence in generated abstractive summaries due to the lack of considerations in context-diversified latent representations of input source texts. To deal with textual representation learning in context-varied situation, pre-trained linguistic embedding frameworks, (e.g., BERT [10]) have been applied and demonstrated dramatic improvements in accuracy performance in which proposed models are fine-tuned for both sufficient context-varied natural language understanding and effective abstractive summary generation task, such as well-known works of ES [12] MNSum [13], DASum [14], etc. Recent works of Song, K. et al. in proposed MASS [16] automatic text generation model has shown the usefulness of using seq2seq-based with masked linguistic mechanism to support text generation task. Similar to the recent proposal of Lewis, M. et al. in BART model [28] with a novel introduction of de-noising auto-encoding mechanism which supports to sample span of texts by the random token masking strategy. Although rich-contextual pre-trained language-based models have achieved remarkable successes in abstractive text summarization, there are existing limitations related to the generation of trivial and generalized summaries. The root cause of these limitations is majorly come from the involving of high-frequency phrase occurrence and common contextual information of same writing styles of documents in the training set. Thus, it leads to the issue of generic summaries are generated for documents with the same writing context.

To deal with this limitation, several attempts [19] [20] [21] which rely on the integration between generative adversarial network (GAN) and reinforcement learning (RL) to deal with the generic summary generation problem. However, recent GAN-RL based models still suffer problems related to unnatural and influent generated summaries due to the lack of thorough evaluations on analyzing contextual and long-range dependent features of texts. Different from recent GAN-RL based abstractive text summarization techniques, in this paper we proposed a novel semantic and long-range structural enhanced representation learning to sufficiently capture all the rich contextual and structural information of words in each training document which are then used to facilitate the summary generation process via the integrated GAN with policy gradient optimization training strategy.

3. SGAN4ABSUM MODEL

In this section, we formally present the methodology of our proposed SGAN4AbSum model which is a text-enhanced GAN-RL based approach for abstractive text summarization task. In the first part of this section, we provide brief descriptions about the problem formulation of abstractive summarization, related background concepts which are used in our paper.

3.1. Preliminaries & background concepts

Generally, abstractive text summarization (definition 1) is considered as an important application of NLP domain which aims to produce short, accurate and fluent summary ($\mathfrak{s}$) for a length-varied textual document ($\mathcal{d}$). In fact, the abstractive text summarization is considered as more challenging than the extractive one due to the requirements of multiple linguistic analysis and understanding process to accurately interpret a length-varied document into a shorter form but still sufficiently carries important information of the whole source text. To fully capture the rich-semantic representations of input texts, recent abstractive summarization models [9] [16] [28] adopt advanced neural encoder-decoder based architecture to assist the processes of text representation learning and generation.

Definition 1: abstractive text summarization (ATS)*. Given a document, denoted as: $\mathcal{d}$ which contains a set of sentences, as: $\mathcal{d} = \{\mathcal{s}_1, \mathcal{s}_2, \dots, \mathcal{s}_{|\mathcal{d}|}\}$ or $\mathcal{d} = \{\mathcal{s}_i\}_i^{|\mathcal{d}|}$, then in each sentence ($\mathcal{s}$), we a set of separated words, as: $\mathcal{s} = \{w_1, w_2, \dots, w_{|\mathcal{s}|}\}$ or $\mathcal{s} = \{w_i\}_i^{|\mathcal{s}|}$. The ultimate objective of an abstractive text summarization model is to learn and generate the corresponding summary as a set of words, denoted as:*

$\mathfrak{s}_d = \{w_i\}_i^{|\mathfrak{s}_d|}$ *for a specific document (*d*. In general, most of abstractive summarization models are designed as a supervised learning approach which aims to learn the parameters of the abstractive summarization mechanism as a mapping function, as:* $f_{ATS}(.)$ *by using a set of training set, denoted as:* $\mathcal{T} = \{\langle d, \mathfrak{s}_d^{gt}\rangle_i\}_{i=1}^{|\mathcal{T}|}$ *with* $\mathfrak{s}_d^{gt}$ *is the ground-truth/golden summary of a given document (*d*). Varied in different models, multiple training architecture and strategies are applied to approximate the* $f_{ATM}(.)$ *to effectively handle the abstractive summarization task, as:* $f_{ATS}(d) \rightarrow \mathfrak{s}_d$.

Definition 2: BERT [10]. *Recently introduced by Devlin, J. et al., BERT is a powerful rich-contextual textual representation framework which enables to fine-tune for multiple tasks in NLP, include the text/summary generation. There are multiple pre-trained versions of BERT for different languages. The pre-trained BERT model is a trained version with multiple large-scale text corpora in order to sufficiently carry out most of different contextual information of words in a specific language. The pre-trained BERT model is defined a textual embedding mapping function, as:* $f_{BERT}(.)$ *which supports to transform words in each sentence (*s*) into the fixed d-dimensional embedding vector, denoted as:* $f_{BERT}(s) \rightarrow \{\overrightarrow{e_1^w}, \overrightarrow{e_2^w}, \dots, \overrightarrow{e_{|s|}^w}\}$.

Among recent advanced textual representation learning techniques, BERT (definition 2) is considered as the most powerful and flexible tool for effectively learn the rich-contextual information of texts. Recently, BERT and its variants are welly applied and fine-tuned to effectively deal with major challenges of abstractive text summarization such as context-varied and out-of-vocabulary summary representation. However, pre-trained BERT model is only capable to capture the local contextual information of words within a sentence, thus it is unable to fully capture the latent long-range dependencies between words at the document's level. Thus, in this paper we tend to use an integrated pre-trained BERT with graph convolutional network (GCN) [24] architecture to fully capture the semantic and long-range structural latent features of words in forms of document's graphs. The learnt rich-semantic word representations are later used to facilitate the process of summary generation via the deep neural encoder-decoder architecture. Table 1 provides a list of notations/mathematic symbols and corresponding descriptions which are commonly used in the left contents of our paper.

Table 1. List of common notations & descriptions

Notation	Description
w, s and d	A single word, sentence and document, respectively.
$\mathfrak{s}_d$ and $\mathfrak{s}_d^{\mathrm{gt}}$	The machine-based generated and ground-truth summaries of a given document (d), respectively.
$\mathcal{G}_d = (\mathcal{V}_d, \mathcal{E}_d)$	The textual graph-based structure of a given document (d) with a set of vertices ($\mathcal{V}$) as unique ocurring words and edges ($\mathcal{E}$) as the n-hop co-occuring relationships beween two words.
$\overrightarrow{\mathrm{e}^w}$, $\overrightarrow{\mathrm{e}^s}$ and $\overrightarrow{\mathrm{e}^d}$	The embedding vectors of a word, sentence and document, respectively.
$\mathrm{X} \in \mathbb{R}^{\mathrm{n}\times\mathrm{d}}$	An d-dimensional embedding matrix with (n) rows.
$\mathcal{H}$	A hidden state vector/matrix.
$\mathrm{FNN}(.)$	A full-connected neural network archiecture as a mapping function.
G and D	The generator and discriminator models, respectively.
Θ	A set of trainable weighting/bias parameters of a model/neural archiecture.

3.2. BERT-GCN-TextEmb: joint semantic and structural text representation learning

First of all, we apply the pre-trained BERT model to learn the contextual representations of all words in each sentence (s), denoted as: $f_{BERT}(s) \rightarrow \{\overrightarrow{e_1^w}, \overrightarrow{e_2^w}, \dots, \overrightarrow{e_{|s|}^w}\}$, then all sentences in each input document (d). For a set of common words which are occurred in multiple sentences of a given document (d), we apply a non-linear fusion mechanism to fully merge different representations of a unique word (w) into a unified embedding space. For a set of (n) sentence-varied d-dimensional word embedding vectors of a unique word (w) in a given document (d), denoted as: $X_w^{BERT,d} = \{\overrightarrow{e_i^w}\}_{i=1}^{n}$ and $X_w^{BERT,d} \in \mathbb{R}^{n \times d}$, we apply a non-linear fusion mechanism to form a unified embedding vector of a unique word (w), denoted as: $\overrightarrow{e^{BERT,w}}$, this process can be generally formulated as the following (as shown in equation 1):

$$Z^{fuse}\left(\overrightarrow{e_i^w}\right) = W_\beta^{fuse}.\overrightarrow{e_i^w} + b^{fuse}$$

$$\overrightarrow{e^{BERT,w}} = f_{fuse}\left(X_w^{BERT,d}\right) = \sigma\left(\sum_{\overrightarrow{e_i^w} \in X_w^{BERT,d}} W_\alpha^{fuse}\, \sigma\left(Dropout\left(Z^{fuse}\left(\overrightarrow{e_i^w}\right)\right)\right)\right) \quad (1)$$

For the set of corresponding parameters of the given fusion mechanism, as: $\Theta^{fuse} = \{W_\alpha^{fuse}, W_\beta^{fuse}, b^{fuse}\}$ are simultaneously optimized during the training process of the given generator which will be described in later section. The unified word embedding vector ($\overrightarrow{e^{BERT,w}}$) which is achieved in the after all is considered as sufficiently carrying all rich-semantic sentence-varied contextual information of each word in the given document (d). The learnt all d-dimensional BERT-based word embedding vectors of given document (d) is presented as a word embedding matrix, denoted as: $X_d^{BERT} \in \mathbb{R}^{|d| \times d}$.

On the other aspect, for the long-range dependent relationships between words, we apply a multi-layered GCN architecture to learn the representations of words over the graph-based structure of a given document (d). To do this, we firstly apply the textual graph-based transformation in each document (d) to build the document's graph, denoted as: $\mathcal{G}_d = (\mathcal{V}_d, \mathcal{E}_d)$ with $\mathcal{V}_d$ is a set of unique words which are occurred in document (d) and $\mathcal{E}_d$ is a set of n-hop co-occurring relationships between two continuous words. Then, we apply a k-layered GCN-based encoder to learn the latent representations of all words as nodes in the given document's graph through GCN-based propagation learning process. For the initial hidden state of our GCN architecture, we use the BERT-based word embedding matrix of the document (d) as the initial node weighting attributes, then the first hidden state out of this initial layer is identified as the following (as shown in equation 2):

$$\mathcal{H}^{GCN,[1]} = ReLu\left(W^{GCN,[0]}.X_d^{BERT}.\mathcal{A}^*\right) \quad (2a)$$

$$\mathcal{H}^{GCN,[l]} = ReLu\left(W^{GCN,[l-1]}.\mathcal{H}^{GCN,[l-1]}.\mathcal{A}^*\right) \quad (2b)$$

Then, at each l^{th} layer, the output hidden state is accordingly identified shown in equation 2b. We apply the same architecture of the original GCN [24] with $\mathcal{A}^*$ is the normalized adjacency matrix of given document's graph ($\mathcal{G}_d$). The normalized adjacency matrix is identified as: $\mathcal{A}^* = \widetilde{\mathcal{D}}^{-\frac{1}{2}}\tilde{\mathcal{A}}\widetilde{\mathcal{D}}^{-\frac{1}{2}}$, where: $\tilde{\mathcal{A}} = \mathcal{A} + \mathcal{J}$ and $\widetilde{\mathcal{D}} = diag\left(\sum_j \tilde{\mathcal{A}}_{ij}\right)$, with: $\mathcal{A}$, $\tilde{\mathcal{A}}$ and $\widetilde{\mathcal{D}}$ are the identity matrix, self-connection-based adjacency matrix and degree matrix of the given $\tilde{\mathcal{A}}$, respectively. Then, at end of the propagation learning process, we apply the max pooling on the output hidden state of the last k^{th} layer of the given GCN architecture to achieve the final structural latent embedding vectors of all words in the given document's graph ($\mathcal{G}_d$), denoted as: $MaxPool\left(\mathcal{H}^{GCN,[k]}\right) \rightarrow X_d^{GCN}$. Finally, to fully fuse the separated BERT-based and GCN-based

word embedding vectors, we reuse the previous non-linear fusion mechanism to effectively learn and merge semantic and long-range dependent latent features of words into a unified embedding space. The overall process for merging BERT-based and GCN-based word embedding matrices, as: $\mathrm{X}_d^{\mathrm{BERT}}$ and $\mathrm{X}_d^{\mathrm{GCN}}$, respectively, to produce the final word embedding matrix (X_d), is defined as the following equation (as shown in equation 3):

$$\begin{aligned} \mathrm{Z}^{\mathrm{fuse}}\left(\overrightarrow{\mathrm{e}_1^{\mathrm{GCN},w}}, \overrightarrow{\mathrm{e}_1^{\mathrm{BERT},w}}\right) &= \mathrm{W}_\beta^{\mathrm{fuse}}.\left[\overrightarrow{\mathrm{e}_1^{\mathrm{GCN},w}}, \overrightarrow{\mathrm{e}_1^{\mathrm{BERT},w}}\right] + \mathrm{b}^{\mathrm{fuse}} \\ \mathrm{X}_d &= \mathrm{f}_{\mathrm{fuse}}\left(\mathrm{X}_d^{\mathrm{GCN}}, \mathrm{X}_d^{\mathrm{BERT}}\right) \\ &= \sigma\left(\sum_{\overrightarrow{\mathrm{e}_1^{\mathrm{GCN},w}} \in \mathrm{X}_d^{\mathrm{GCN}}} \sum_{\overrightarrow{\mathrm{e}_1^{\mathrm{BERT},w}} \in \mathrm{X}_d^{\mathrm{BERT}}} \mathrm{W}_\alpha^{\mathrm{fuse}}\, \sigma\left(\mathrm{Dropout}\left(\mathrm{Z}^{\mathrm{fuse}}\left(\overrightarrow{\mathrm{e}_1^{\mathrm{GCN},w}}, \overrightarrow{\mathrm{e}_1^{\mathrm{BERT},w}}\right)\right)\right)\right) \end{aligned} \tag{3}$$

Similar to the previous fusion mechanism which is used to merge BERT-based word embedding vectors, all parameters of this BERT+GCN-based fusion mechanism are also jointly optimized with the generator model which will be described right after this section. At the end of this textual representation learning process, we achieve a set of unified semantic and structural long-range dependent latent representations of words in each source document which are later used to assist the summary generation process in the generator component.

3.3. Generator & Discriminator of SGAN4ABSUM model

Neural encoder-decoder based generator model. Similar to recent studies [9] [19] [20] on abstractive text summarization with the integrated GAN and reinforcement learning approach, we also apply an attention-based neural encoder-decoder architecture in our generator model for handling the sequential representation learning of input texts and corresponding summary generation. For the encoder, it receives the sequence of BERT-GCN-TextEmb-based word embedding vectors which have been achieved in previous steps as the inputs and feed them into a Bi-LSTM architecture to generate corresponding output hidden states in both forward and backward directions, denoted as: $\mathcal{H}^{\mathrm{enc},+}$ and $\mathcal{H}^{\mathrm{enc},-}$, respectively. Then, we concatenate the output hidden states of the given Bi-LSTM architecture to produce the final encoder's output, as: $\overrightarrow{e^{enc}}$. At the RNN-based decoder side, for each t^{th} time-step, it receives the concatenated hidden states of encoder's Bi-LSTM architecture as the inputs and combine them with its current hidden state, denoted as: $\mathcal{H}_{[\mathrm{t}]}^{\mathrm{dec}}$ to produce the context vector, denoted as: $\mathcal{C}_{[\mathrm{t}]}^{\mathrm{dec}}$ by identifying the additive attention distribution [4] and weighted sum scores of the encoder's inputs. All processes in the given encoder-decoder architecture can be formulated as the following (as shown in equation 4):

$$\begin{aligned} \overrightarrow{e^{\mathrm{enc}}} &= [\mathcal{H}^{\mathrm{enc},+}, \mathcal{H}^{\mathrm{enc},-}] \\ \alpha_{\mathrm{t}}^{\mathrm{Att}} &= \mathrm{softmax}\left(\mathcal{v}^{\mathrm{T}}\tanh\left(\mathrm{W}^{\mathrm{enc,Att}}.\overrightarrow{e^{\mathrm{enc}}} + \mathrm{W}^{\mathrm{dec,Att}}.\mathcal{H}_{[\mathrm{t}]}^{\mathrm{dec}} + \mathrm{b}^{\mathrm{Att}}\right)\right) \\ \mathcal{C}_{[\mathrm{t}]}^{\mathrm{dec}} &= \sum \alpha_{\mathrm{t}}^{\mathrm{Att}}.\overrightarrow{e^{\mathrm{enc}}} \end{aligned} \tag{4}$$

Then, the context vector ($\mathcal{C}_{[\mathrm{t}]}^{\mathrm{dec}}$) is combined with the current state ($\mathcal{H}_{[\mathrm{t}]}^{\mathrm{dec}}$) of the given RNN-based decoder at a specific t^{th} time-step and feed to a full-connected neural network architecture with two linear dense layers and a softmax classification function at the end to compute the probabilistic distribution of all words in the vocabulary, denoted as: $\mathrm{Prob}^{\mathcal{W}}$. This process can be simply formulated as the following (as shown in the equation 5):

$$\begin{aligned} \overrightarrow{e^{\mathrm{enc}}} &= \left[\mathcal{C}_{[\mathrm{t}]}^{\mathrm{dec}}, \mathcal{H}_{[\mathrm{t}]}^{\mathrm{dec}}\right] \\ \mathrm{FNN}\left(\overrightarrow{e^{\mathrm{enc}}}\right) &= \mathrm{W}_\alpha^{\mathrm{FNN}}.\mathrm{Dropout}\left(\mathrm{W}_\beta^{\mathrm{FNN}}.\overrightarrow{e^{\mathrm{enc}}} + \mathrm{b}_\beta^{\mathrm{FNN}}\right) + \mathrm{b}_\alpha^{\mathrm{FNN}} \end{aligned} \tag{5}$$

$$\text{Prob}^{\mathcal{W}} = \text{softmax}\big(\text{FNN}(\overrightarrow{e^{\text{enc}}})\big)$$

In general, the computed $\text{P}^{\mathcal{W}}$ is a probability distribution of all words in the vocabulary, thus also provide the appearance probabilistic distribution of each word (w) in the generated summary, as: s_d, for a input document (d), denoted as: $\text{Prob}(w) = \text{Prob}^{\mathcal{W}}$ To effectively deal with the OOV-related challenge in abstractive text summarization, we reapply the pointer-generator network of the GTTP model [9].

Abstractive text summarization task-driven discriminator model. Majorly inherited from previous GAN-RL based models [19] [20], our discriminator is designed to work as a binary classifier which is in charge of distinguishing a generated sequential words is a human written summary or machine-based generated one. In order to do this, we firstly apply the BERT-GCN-TextEmb-based textual embedding mechanism to learn the rich-semantic and structural representations of all words in a given summary, as: X_{s}. Then, the learnt word embedding matrix is applied vertical max pooling strategy to form the final embedding vector of the given summary, denoted as: $\text{MaxPool}(\text{X}_{\text{s}}) \rightarrow \overrightarrow{\text{e}^{\text{s}}}$. Then, the BERT-GCN-TextEmb-based representation of $\overrightarrow{\text{e}^{\text{s}}}$ is fed to a full-connected neural network with 1 linear dense layer and the sigmoid activation function at the end to handle binary classification task. The final output of this full-connected neural architecture is the probabilistic distribution (Prob^{s}) of the given summary is human written (labelled as: 1) or not (labelled as: 0). In general, the overall architecture of our discriminator model can be illustrated as the following (as shown in equation 6):

$$\begin{aligned} \text{FNN}\big(\overrightarrow{\text{e}^{\text{s}}}\big) &= \text{W}^{\text{FFN}}.\overrightarrow{\text{e}^{\text{s}}} + \text{b}^{\text{FNN}} \\ \text{Prob}^{\text{s}} &= \sigma\big(\text{FNN}(\overrightarrow{\text{e}^{\text{s}}})\big) \end{aligned} \tag{6}$$

3.4. Policy gradient training strategy

To efficiently learn and optimize overall model's parameters, we apply the policy gradient strategy which is inherited from previous works [25] [26] to maximize the cumulative total reward of the generator can achieve at each step after generating a summary. The learning objective is formally formulated as the following (as show in equation 7), with: $\text{s}_{[1:|\text{s}|-1]} = \{w_1, w_2, \dots, w_{|\text{s}|-1}\}$:

$$\mathbb{E}_{\text{G}_\Theta}[\text{R}] = \sum_{w \in \text{s}} \text{G}_\Theta(\text{s}|d).\,\text{Q}_{\text{D}_\phi}^{\text{G}_\Theta}\big(\text{s}_{[1:|\text{s}|-1]}, d, \text{s}_{[-1]}\big) \tag{7}$$

With Θ is the set of parameters of the given generator model which are optimized by performing the gradient descent on the $\mathbb{E}_{\text{G}_\Theta}[\text{R}]$. The $\text{Q}_{\text{D}_\phi}^{\text{G}_\Theta}(.)$ is the action-value function in which the state is the generated summary. As mentioned above, the discriminator in our proposed architecture plays as a binary classifier which supports to identify the summary is the human written or not in forms of a probabilistic distribution. Thus, this probability is considered as the reward for our generator, formulated as: $\text{Q}_{\text{D}_\phi}^{\text{G}_\Theta}\big(\text{s}_{[1:|\text{s}|-1]}, d, \text{s}_{[-1]}\big) = \text{D}(\text{s}, d) - \text{b}(\text{s}, d)$. Then, the n-time Monte Carlo search strategy is applied to sample unknown words from the generated summary (s) in comparing with the ground-truth one (s^{gt}) of a given document (d). From that, we can achieve the (n) rewards for each state and take the average result as the final reward. Strictly following the original training strategy of GCN, we re-train the discriminator after it receives the generated summary from the generator. For both generator and discriminator, the training objectives are formulated as the following (as shown equation 8):

$$\begin{aligned} \min\text{-}\mathbb{E}_{\langle d, \text{s}_d^{\text{gt}} \rangle \in \mathcal{T}} &= [\log \text{D}(\text{s}, d)] - \mathbb{E}_{\langle d, \text{s}_d \rangle \in \text{G}_\Theta}\big[\log\big(1 - \text{D}(\text{s}, d)\big)\big] \\ \nabla \mathbb{E}_{\text{G}_\Theta}[\text{R}] &= \frac{1}{|\text{s}|}\sum_{\text{i}=1}^{|\text{s}|} \mathbb{E}_{w_\text{i} \in \text{G}_\Theta} \nabla \log \text{G}_\Theta\big(w_\text{i} \big| \text{s}_{[1:|\text{s}|-1]}, d\big).\,\text{Q}_{\text{D}_\phi}^{\text{G}_\Theta}\big(\text{s}_{[1:|\text{s}|-1]}, d, w_\text{i}\big) \end{aligned} \tag{8}$$

4. EXPERIMENTS & DISCUSSIONS

To evaluate the effectiveness of our proposed SGAN4ABSUM model, in this section we provide extensive experiments for the abstractive text summarization task in the CNN/Daily-Mail benchmark dataset. The experimental results in terms of ROUGE-based metric demonstrate the outperformances our proposed ideas in this paper in comparing with recent abstractive summarization baselines.

4.1. Dataset description & experimental setups

4.1.1. Dataset & textual pre-processing steps

The CNN/Daily-Mail dataset [29] [30]. This is considered as a common dataset for abstractive text summarization task which contains about > 300K pairs of news' contents and the corresponding human written abstractive summary. The CNN/Daily-Mail dataset contains three parts, includes: training, testing, validation sets. Table 2 shows general statistics about the used CNN/Daily-Mail dataset for all experiments in our paper. The extra information and pre-processing scripts for CNN/Daily-Mail dataset can be achieved at this repository[1].

Table 2. General statistic about the CNN/Daily-Mail dataset

Dataset's Information	Value
Training set (pairs)	287,113
Testing set (pairs)	11,940
Validation set (pairs)	13,368
Average document length (after pre-processing steps)	781.25

Text pre-processing steps & experimental configurations. We followed the pre-processing steps of previous works limit the size of input source documents and the corresponding human-written summaries to 800 and 100, respectively. For textual pre-processing steps for constructing the text-graph of each input document, such as: word tokenization, word stemming, sentence splitting, etc. we utilized the well-known open-source Stanford-NLP library[2] [31]. The document's graphs are constructed with the set of 2-hop co-occurring relationships between unique words in each document. For the pre-trained BERT model, we used the original English pre-trained BERT (large/uncased) version which is available to achieve at this repository [3]. In the setups of our proposed BERT-GCN-TextEmb textual embedding mechanism, for the configuration of BERT-based word and document embedding dimensionality, we all set the dimensional size of these embedding vectors to 400. For the setup of GCN-based structural textual encoding mechanism over constructed document's graph, we implemented the original GCN architecture of Kipf, T. N. et al. [24] with number of GCN-based layer is set to 5. At the generator side, for the inside neural encoder-decoder architecture, we set the number of used LSTM-based cells for both Bi-LSTM-based encoding and LSTM-based decoding as 300. The detailed configurations for our proposed SGAN4AbSum model can be found in Table 3.

Table 3. Configurations for our proposed SGAN4AbSum model for all experiments in this paper

Configuration parameter	Value
Dimensionality of BERT-GCN-TextEmb-based word embedding vector, as: (d^{w})	400
Dimensionality of BERT-GCN-TextEmb-based document embedding vector, as: (d^{d})	400

[1] CNN/Daily-Mail dataset: https://github.com/abisee/cnn-dailymail
[2] CoreNLP library for NLP (Java): https://stanfordnlp.github.io/CoreNLP/
[3] Pre-trained BERT (large, uncased): https://github.com/google-research/bert

Number of used GCN-based layers in BERT-GCN-TextEmb textual embedding mechanism	5
Number of used LSTM cells for the neural encoder-decoder architecture in the generator model	300
Number of training epochs for the GAN-based architecture	80
General learning rate for all neural network architectures in SGAN4AbSum model	0.001
Number of training batch size	64

4.1.2. Evaluation metric usage

In order to evaluate the experimental outputs of the abstractive text summarization task with different baselines, we mainly used the ROUGE-based standard metric [32] with ROUGE-1, ROUGE-2 and ROUGE-L scores. In general, the ROUGE-based evaluation method assesses the accuracy performance of a given abstractive text summarization model by utilizing the n-grams (unigram, bigram, etc.) overlapping evaluation upon the achieved information of the generated summaries in comparing with the ground-truth data. The ROUGE-based score for general n-gram (ROUGE-N) and longest common sequence (ROUGE-L) are identified as the following (as shown in the equation 9 & 10) [32]:

$$\text{ROUGE-N}=\frac{\sum_{s\in\mathcal{S}}\sum_{\text{gram}_n\in s}\text{Count}_{\text{match}}(\text{gram}_b)}{\sum_{s\in\mathcal{S}}\sum_{\text{gram}_n\in s}\text{Count}(\text{gram}_b)} \tag{9}$$

$$\text{ROUGE-L}=\frac{(1+\beta^2)\frac{\text{LCS}(X,Y)}{m}.\frac{\text{LCS}(X,Y)}{n}}{\frac{\text{LCS}(X,Y)}{m}+\beta^2.\frac{\text{LCS}(X,Y)}{n}} \tag{10}$$

In the original descriptions of Lin, C. Y. [32], the $\mathcal{S}$ and s, are the set of ground-truth summaries and the given generated summary with $\text{Count}_{\text{match}}$ and LCS are the maximum value in which the given n-grams co-occurring in a generated summary and a set of ground-truth summaries and length of longest common sequence between two input texts. In case the value $\text{LCS}(X,Y)=0$, the ROUGE-L $=0$ and it is equal to 1 in case X = Y.

4.1.3. Comparative abstractive summarization baselines

To evaluate the effectiveness of our proposed SGAN4AbSum model in comparing with other baselines, we implemented several well-known abstractive summarization model for the comparative purpose, which are:

- **Seq2SeqAbSum** (2016) [7]: is considered as an early integrated attention seq2seq-based approach for the abstractive text summarization. In this model, Nallapati, R. et al. proposed an approach of using an attention-based encoder-decoder architecture to handle abstractive text summarization in which the mechanism is similar to previous neural machine translation architectures [3] [4]. At the decoder side, Nallapati, R. et al. proposed the use of multiple-levelled (word/sentence)-level attention mechanism to effectively handle the summary generation process.
- **SummaRuNNer** (2017) [8]: is also an seq2seq-based text generation model for both extractive and abstractive text summarization tasks. In SummaRuNNer model, Nallapati, R. et al. proposed a dynamic sentence-level representation learning technique which enables the system to flexibly utilize for different training objectives. For handling abstractive summarization task, we followed the guidance of Nallapati, R. et al. for abstractive text summarization [8] in the original work with the RNN-based encoder-decoder implementation.
- **GTTP** (2017) [9]: is considered as a recent well-known seq2seq-based abstractive text summarization model which utilizes a novel pointer-generator network with soft attention-based mechanism. In the GTTP model, See, A. et al. proposed the utilization of soft attention-based mechanism [9] in the pointer-generator network to effectively capture and produce the desired

output summaries which sufficiently contains the salient information of the input source texts. Experiments in CNN/Daily-Mail benchmark dataset demonstrate the effectiveness of GTTP in abstractive summarization task.

- **DeepRLAbSum** (2018) [23]: is an extensive RNN-based encoder-decoder architecture for abstractive summarization with the novel application of using reinforcement learning (RL) for the training strategy optimization. In the DeepRLAbSum, Paulus, R. et al. [23] proposed a novel seq2seq-based architecture with the custom intra-attention mechanism to let the model much focuses on the input source texts and the corresponding continuously generated outputs. The used seq2seq-based architecture of DeepRLAbSum are jointly trained with the combination of classical supervised reinforcement learning (RL)-based strategies.
- **GAN-RL** (2018) [19]: is considered as an early GAN-based abstractive summarization approach which utilizes the adversarial network training strategy to effectively deal with the unnatural representation of generated summaries. In the GAN-RL model, Liu, L. et al. [19] proposed the use of RNN-based seq2seq model with attention mechanism in the generator to handle the text generation task, then the generated summaries are evaluated by the discriminator to identify the input summaries are human-written ones or not. Then, the parameters of generator and discriminator are jointly optimized by using the policy gradient training strategy of RL approach. Through experiments in benchmark CNN/Daily-Mail dataset, GAN-RL model demonstrates remarkable improvements in abstractive summarization task.
- **GAN-RL-TDA** (2019) [20]: is similar to GAN-RL model [19] which utilizes the integrated adversarial network architecture and policy gradient training strategy of RL to handle abstractive summarization task. In GAN-RL-TDA model, Rekabdar, B. et al. [20] applied the time-decay based attention mechanism to leverage the quality of generated summaries.

For above listed comparative baselines, we implemented them with the same configurations in which these models achieve the highest accuracy performance as described in their original published papers. For common configurations which are similar to our proposed SGAN4AbSum model, we configured them the same as described in Table 3.

4.2. Experimental results & discussions

Table 4 shows the experimental outputs in terms of ROUGE-1, ROUGE-2 and ROUGE-L metrics for abstractive text summarization task in the standard CNN/Daily-Mail dataset. As shown from the experimental outputs, our proposed SGAN4AbSum explicitly outperforms all the baselines for all ROUGE-based metrics, include ROUGE-1, ROUGE-2 and ROUGE-L.

Table 4. Experimental outputs for abstractive summarization task in terms of ROUGE-1, ROUGE-2 and ROUGE-L metrics by different baselines in the benchmark CNN/Daily-Mail dataset

	ROUGE-1		**ROUGE-2**		**ROUGE-L**	
Seq2SeqAbSum	35.481		14.622		31.891	
SummaRuNNer	38.216	↑0.077	16.982	↑0.161	35.278	↑0.106
GTTP	39.545	↑0.115	18.023	↑0.233	37.662	↑0.181
DeepRLAbSum	40.692	↑0.147	19.281	↑0.319	38.668	↑0.213
GAN-RL	39.882	↑0.124	20.289	↑0.388	36.592	↑0.147
GAN-RL-TDA	40.261	↑0.135	19.269	↑0.318	37.342	↑0.171
SGAN4AbSum	**41.827**	↑0.179	**21.663**	↑0.482	**39.021**	↑0.224

In general, it is quite clear from the experimental outputs that most of GAN-RL based abstractive text summarization approach (GAN-RL, GAN-RL-TDA and our proposed SGAN4AbSum) slightly achieve better performance than the traditional neural seq2seq-based approach (Seq2SeqAbSum, SummaRuNNer,

GTTP and DeepRLAbSum) approximately 5.64%, 18.45% and 4.95% in terms of ROUGE-1, ROUGE-2 and ROUGE-L metrics, respectively. It proves the usefulness of applying adversarial network training strategy in the text summarization task in which the generated summaries are automatically controlled by the discriminator and jointly optimized with the RNN-based textual generation mechanism in generator. It supports to not only improve the quality but also the natural form of generated summaries in the after all. In more details, our proposed SGAN4AbSum significantly achieved better performance than previous seq2seq-based models averagely 8.38%, 25.75% and 8.77% in terms of ROUGE-1, ROUGE-2 and ROUGE-L, respectively. For our main competitors, which are: GAN-RL and GAN-RL-TDA model, it also slightly leverages the accuracy performance averagely 5.94% and 5.82%, respectively.

In the after all, our extensive experiments in the CNN/Daily-Mail dataset demonstrate the effectiveness of our proposed ideas in this paper. The competitive experimental results in terms of ROUGE-based metrics prove the usefulness of our text-enhanced GAN-RL based text abstractive summarization approach. By using an integrated BERT+GCN based textual representation learning approach, called as: BERT-GCN-TextEmb, which can support to effectively capture both semantic and structural latent features of the input documents. Then, the rich-semantic representations of input texts are used to facilitate the summary generation process of the generator model.

4.3. Ablation studies

In this section, we demonstrated experimental studies related to our model's parameter sensitivity analysis. Originally as a neural textual representation learning approach, thus there are vital model's parameters which might be sensitive and need to be taken in consideration for practical implementation, such as: dimensionality of word and document embedding vectors, number of used GCN-based layer (in our proposed BERT-GCN-TextEmb mechanism), number of used LSTM-based cells in the encoder-decoder architecture and the number of training iterations.

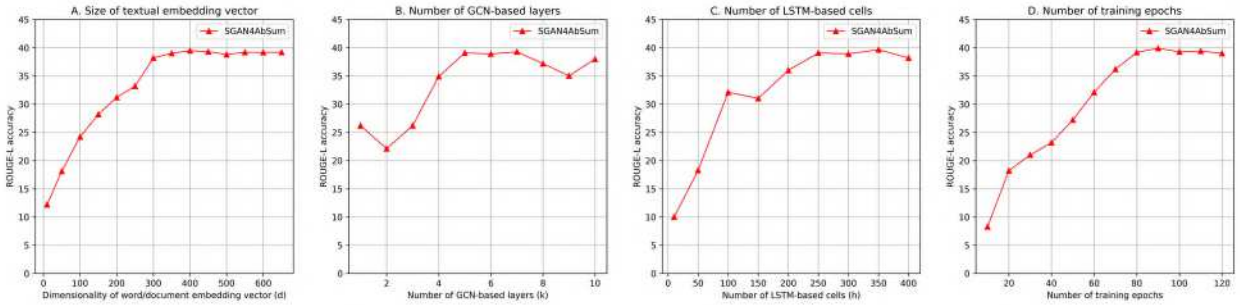

Figure 2. Experimental studies on the parameter sensitivity of our proposed SGAN4AbSum model

Dimensionality of word and document embedding vector. To study the influence of the dimensionality of word/document embedding vector as: (d), we varied the value of (d) parameter from 10 to 600 and reported the changes of model's accuracy performance in terms of ROUGE-L score. As shown from the experimental outputs in Figure 2-A, our proposed SGAN4AbSum achieved the highest performance with value of (d) parameter is over 350 which proves that our proposed SGAN4AbSum model is quite insensitive with this parameter in which a length-varied dataset need a reasonable value of dimensionality embedding vector to sufficiently capture all features of the given input texts.

Number of GCN-based layers. In our proposed BERT-GCN-TextEmb-based textual representation learning approach, we apply a GCN-based architecture to capture the long-range dependent structural features of input source texts. As a multi-layered graph-based neural architecture, the number of GCN-based layers, as: (k) also can affect the overall model's accuracy performance. To study the influence of this parameter, we also varied the value of (k) in range [1, 10] and investigated the changes in ROUGE-L based accuracy outputs of our SGAN4AbSum model. As shown in Figure 2-B, our proposed model achieved the highest performance with the value of (k) parameter is in range [5, 7], whereas it encounter

the light oscillations with other values of (k). Thus, this experimental outputs show that our proposed BERT-GCN-TextEmb-based textual embedding strategy is quite sensitive with the number of used GCN-based layers which must be carefully considered in practical implementation.

Number of used LSTM-based cells and training epochs. As a neural encoder-decoder architecture, the seq2seq-based generator is considered as the main component of our proposed SGAN4AbSum model which is implemented by two LSTM-based encoding and decoding mechanisms to fulfill the text generation task. To study the effects of number of used LSTM-based cells, denoted as: (h) in this component, we also altered the value of (h) parameter in range [10, 400] and reported the changes in ROUGE-L based accuracy results. As shown from the experimental outputs in Figure 2-C, our model is quite insensitive with this parameter and reach the highest performance with value of (h) is over 250. For the training process and performance optimization of the SGAN4AbSum model, it can be clearly seen from the Figure 2-D that our proposed model reach the convergent point after 80 training epochs which is a reasonable number of training iterations in cases of handling large-scale datasets with efficiency in computational cost and time-consuming performance.

5. CONCLUSIONS & FUTURE WORKS

Among primitive tasks of NLP domain, the abstractive text summarization is considered as a challenging task in particular when the given input documents are represented as long, complex structural and context-varied forms. It leads to challenges related to the generation of unnatural/influent summaries of seq2seq-based models. Thus, to deal with these challenges, in this paper we proposed a novel text-enhanced GAN-RL based abstractive summarization technique, called as: SGAN4AbSum. In our proposed SGAN4AbSum model, we introduce the use of integrated pre-trained BERT and GCN to jointly capture the rich-semantic and long-range dependent structural features of input texts. Then, the learnt rich-semantic textual representations of input documents are used to facilitate the text interpretation and generation processes of the seq2seq-based generator model. Following the recent approaches of combining adversarial neural network and policy gradient training strategy of RL, we jointly optimize the parameters of generator and discriminator in our proposed SGAN4AbSum model. Extensive experiments in CNN/Daily-Mail benchmark dataset demonstrate the effectiveness of our proposed model in comparing with recent state-of-the-art abstractive text summarization baselines. For our future works, we intend to incorporate the knowledge graph/expert linguistic knowledge to the textual representation learning process of BERT-GCN-TextEmb mechanism to improve the quality in both information and fluency of the generated summaries.

DECLARATIONS

This study was funded by Thu Dau Mot University, Binh Duong, Vietnam.

ACKNOWLEDGEMENT

This research is funded by Thu Dau Mot University, Binh Duong, Vietnam.

CONFLICT OF INTEREST

This research is funded by Thu Dau Mot University, Binh Duong, Vietnam.

REFERENCES

[1] El-Kassas, W. S., Salama, C. R., Rafea, A. A., & Mohamed, H. K., "Automatic text summarization: A comprehensive survey," *Expert Systems with Applications,* p. 113679.

[2] Gambhir, M., & Gupta, V., "Recent automatic text summarization techniques: a survey," *Artificial Intelligence Review,* vol. 47, no. 1, pp. 1-66, 2017.

[3] Sutskever, I., Vinyals, O., & Le, Q. V., "Sequence to Sequence Learning with Neural Networks," in *Advances in Neural Information Processing Systems*, 2014.

[4] Bahdanau, D., Cho, K., & Bengio, Y., "Neural machine translation by jointly learning to align and translate," *3rd International Conference on Learning Representations, ICLR,* 2015.

[5] Rush, A. M., Chopra, S., & Weston, J., "A Neural Attention Model for Abstractive Sentence Summarization," in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, 2015.

[6] Chopra, S., Auli, M., & Rush, A. M., "Abstractive sentence summarization with attentive recurrent neural networks," in *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2016.

[7] Nallapati, R., Zhou, B., Gulcehre, C., & Xiang, B., "Abstractive Text Summarization using Sequence-to-sequence RNNs and Beyond," in *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, 2016.

[8] Nallapati, R., Zhai, F., & Zhou, B., "Summarunner: A recurrent neural network based sequence model for extractive summarization of documents," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2017.

[9] See, A., Liu, P. J., & Manning, C. D., "Get To The Point: Summarization with Pointer-Generator Networks," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, 2017.

[10] Devlin, J., Chang, M. W., Lee, K., & Toutanova, K., "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*, 2019.

[11] Vaswani, Ashish, et al., "Attention is all you need," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, 2017.

[12] Narayan, S., Cohen, S. B., & Lapata, M., "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*, 2018.

[13] Fabbri, A. R., Li, I., She, T., Li, S., & Radev, D., "Multi-News: A Large-Scale Multi-Document Summarization Dataset and Abstractive Hierarchical Model," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019.

[14] Cohan, A., Dernoncourt, F., Kim, D. S., Bui, T., Kim, S., Chang, W., & Goharian, N., "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*, 2018.

[15] Dong, L., Yang, N., Wang, W., Wei, F., Liu, X., Wang, Y., ... & Hon, H. W., "Unified Language Model Pre-training for Natural Language Understanding and Generation," in *33rd Conference on Neural Information Processing Systems (NIPS)*, 2019.

[16] Song, K., Tan, X., Qin, T., Lu, J., & Liu, T. Y., "Mass: Masked sequence to sequence pre-training for language generation," in *International Conference on Machine Learning*, 2019.

[17] Rothe, S., Narayan, S., & Severyn, A., "Leveraging Pre-trained Checkpoints for Sequence Generation Tasks," *Transactions of the Association for Computational Linguistics,* vol. 8, pp. 264-280, 2019.

[18] Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., ... & Bengio, Y., "Generative adversarial networks," in *Proceedings of the 27th International Conference on Neural Information Processing Systems*, 2014.

[19] Liu, L., Lu, Y., Yang, M., Qu, Q., Zhu, J., & Li, H., "Generative adversarial network for abstractive text summarization," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2018.

[20] Rekabdar, B., Mousas, C., & Gupta, B., "Generative adversarial network with policy gradient for text summarization," in *2019 IEEE 13th international conference on semantic computing (ICSC)*, 2019.

[21] Yang, M., Wang, X., Lu, Y., Lv, J., Shen, Y., & Li, C., "Plausibility-promoting generative adversarial network for abstractive text summarization with multi-task constraint," *Information Sciences,* vol. 521, pp. 46-61, 2020.

[22] Yang, M., Qu, Q., Tu, W., Shen, Y., Zhao, Z., & Chen, X., "Exploring human-like reading strategy for abstractive text summarization," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2019.

[23] Paulus, R., Xiong, C., & Socher, R., "A Deep Reinforced Model for Abstractive Summarization," in *International Conference on Learning Representations (ICLR)*, 2018.

[24] Kipf, T. N., & Welling, M., "Semi-supervised classification with graph convolutional networks," in *International Conference on Learning Representations (ICLR)*, 2017.

[25] Yu, L., Zhang, W., Wang, J., & Yu, Y., "Seqgan: Sequence generative adversarial nets with policy gradient," in *Proceedings of the AAAI conference on artificial intelligence*, 2017.

[26] Yang, Z., Chen, W., Wang, F., & Xu, B., "Improving Neural Machine Translation with Conditional Sequence Generative Adversarial Nets," in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2018.

[27] Merity, S., Xiong, C., Bradbury, J., & Socher, R., "Pointer sentinel mixture models," in *Proceedings of the International Conference on Learning Representations*, 2017.

[28] Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., ... & Zettlemoyer, L., "BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020.

[29] Hermann, K. M., Kočiský, T., Grefenstette, E., Espeholt, L., Kay, W., Suleyman, M., & Blunsom, P., "Teaching machines to read and comprehend," in *Proceedings of the 28th International Conference on Neural Information Processing Systems*, 2015.

[30] Nallapati, R., Zhou, B., dos Santos, C., Gülçehre, Ç., & Xiang, B., "Abstractive Text Summarization using Sequence-to-sequence RNNs and Beyond," in *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, 2016.

[31] Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J. R., Bethard, S., & McClosky, D., "The Stanford CoreNLP natural language processing toolkit," in *Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations*, 2014.

[32] Lin, C. Y., "Rouge: A package for automatic evaluation of summaries," in *Text summarization branches out*, 2004.