IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2022.Doi Number

Two-stage text summary model

YANG CHEN[1,2], Roben A. Juanatas[1]

[1]College of computing and Information Technologies, National University, Manila 1008, Philippines (e-mail: chenyang@axhu.edu.cn, rajuanatas@national-u.edu.ph)

[2]Big Data and Artificial Intelligence College, Anhui Xinhua University, Hefei, 230088, China (e-mail: chenyang@axhu.edu.cn)

Corresponding author: Chen Yang (e-mail: chenyang@axhu.edu.cn)

This work was supported in by Anhui Province quality engineering project (2020mooc189) and Anhui Xinhua University quality engineering project (2020kfkcx01&2023jgkcx09).

ABSTRACT In order to solve the problems of redundant information processing and high quality summary generation in existing methods, this paper proposes a two-stage text summary model which is composed of abstracted and generated models. First of all, he important information is abstracted by using an abstracted model which incorporates dilated convolution and gated convolution. Then, a replication mechanism is incorporated into the generated model to ensure that both primary and secondary information are taken into consideration, while also optimizing the cluster search algorithm. Finally, the network structure is reconfigured in the generated model to effectively integrate the coding capabilities of the two-way language model and the text generation abilities of the one-way language model. The experimental findings demonstrate that the two-stage text summary model's performance has been significantly enhanced.

INDEX TERMS Abstracted summary model, BERT, Two-way language model, Generated summary model, One-way language model,

I. INTRODUCTION

The text summary model has grown in significance as a result of the Internet's exponential rise in text content and the abundance of archives containing news items, academic papers, legal documents, and other types of content. When confronted with extensive texts, manual text summarization may become impractical due to human resource costs and time limitations. Scholars have been striving to improve text summarization methods since the 1950s.The methods of text summarization can be categorized into abstraction and generation. Abstraction involves directly abstracting the central semantic sentences from the original text based on specific constraints, and then reordering them to form a summary. The generative form is to generate text summary on the basis of semantic understanding and reconstruction of the original content [1-3].

The technology of abstracted text summary was initially proposed by Luhn et al.[4] in 1958, which automatically identifies keywords in documents based on word frequency and inverse document frequency. It then evaluates the importance of each sentence by combining the number and similarity of keywords within it, resulting in abstracts composed of highly important sentences. This mechanism is used in the existing method. Each sentence in the text is initially assigned a score based on its significance. Subsequently, the sentences are organized in descending order of their scores, and ultimately, those with the highest score and minimal redundancy are selected to form a summary. Sentence position, word frequency, and word chain are just a few examples of the statistical and linguistic factors typically considered when determining sentence value. The process of sentence abstraction can be categorized into two main approaches: supervised and unsupervised. The examples of unsupervised techniques include centroid-based methods[5], graph model-based approaches[6-7], and LDA subject model-based methodologies[8]. Supervised methods include support vector regression [9] and conditional random fields [10].

Generative summaries are more similar to the natural way of human beings to write summaries, which is based on the semantic understanding of the text and uses generative algorithms to summarize and summarize. Neural network models have shown promising results in specific generative summary tasks in recent years. Rush et al.[11] originally presented an attention-mechanic-based encoder and a neural network model decoder model for generative summary tasks, which were influenced by the study on neural machine translation (NMT)[12]. Later, Chopra et al. [13] extended Rush et al. 's work on the recurrent neural

network model of attention mechanism. To improve the results, Nallapati et al.[14] incorporated several techniques into the recurrent neural network-based sequence-to-sequence model, such as implementing word or table constraints in the decoder stage and introducing a hierarchical attention mechanism. A neural network model utilizing the graph attention mechanism was proposed by Tan et al.[15–16] for summary generation, followed by the introduction of a fine-grained approach to title generation. The seq2seq approach was utilized for summary generation by Hu et al.[17], while Gu et al.[18] proposed the integration of a replication mechanism into the seq2seq process for generating summaries. The replication mechanism has two main benefits: first, it can effectively preserve the important information found in the source text through replication; second, it can also be thought of as a combination of abstracted and generated summaries because the output side can generate some summaries that differ in wording from the original text. However, a major limitation of the replication mechanism is that it copies the input information unchanged and cannot be flexibly adjusted. In parallel, See et al.[19] proposed a pointer generator network for summary generation, which effectively addressed the issue by automatically selecting the necessary words to reproduce the summary from the original text or by utilizing the pointer to generate new words from thesaurus. Wang et al.[20] subsequently proposed a bidirectional selective coding model (BiSET) that can effectively abstract important information from each source article to guide the generation of summary using the template identified from the training data. In order to address the issue of producing repetitive words, Babu et al.[21] created a sequence-to-sequence text summary model that combined a time attention mechanism, coverage mechanism, and pointer generator network. Vo[22] proposes a text summary method based on semantic enhanced generative adversarial network (GAN), which uses adversarial training strategies to solve the problems of unnatural and incoherent generated summary. During this period, BART model[23] and T5 (Text-to-text Transfer Transformer) model[24] regard the generation of Text summary as a text-to-text conversion problem, and also achieve good results. There are also some improvements in attention mechanism, optimization methods and embedding of original text information[25-28].

However, there are three main contradictions in the existing text summary models based on neural networks: one is the contradiction between the excessively long text input and the fixed upper limit of the input of the pre-trained model; the other is the contradiction of exposure bias in the text generation process; the third is that the text summary is a text generation task, and the advantages of the two-way language model and the one-way language model are not good for direct integration. Based on this, this paper proposes a two-stage model to solve these problems. Firstly, significant information is abstracted using the abstracted model, which allows the original text's input to be condensed without losing its essential content. Then fill generation mechanism and noise perception generated method are introduced to make up the difference between training and inference. Finally, in order for the generated model to successfully integrate the text generation capability of the one-way language model and the coding capacity of the two-way language model, the network structure is finally reconfigured.

II. TWO-STAGE TEXT SUMMARY MODEL

Fig. 1 illustrates the general layout of the two-stage text summary model that is suggested in this paper. The model consists of two parts: the abstracted summary model based on clause coding and the unidirectional generated summary model based on bidirectional coding. The original text data first obtains the preliminary summary result through the abstracted summary model, and then inputs the preliminary summary result into the generated summary model, and further refines the summary to obtain the final summary generated by the model.

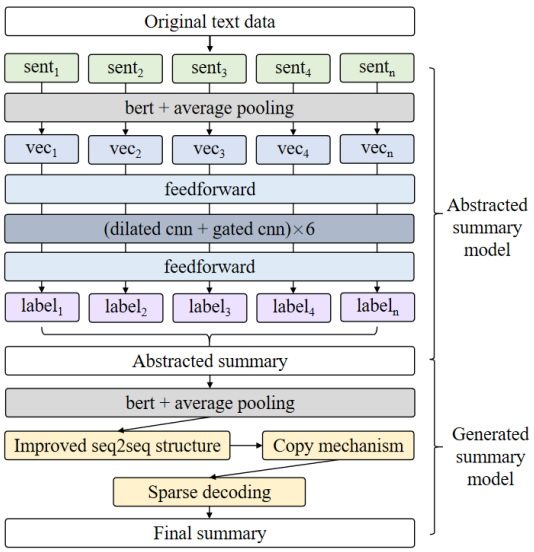

FIGURE 1. Two-stage text summary model

A. ABSTRACTED SUMMARY MODEL

The primary issue that the abstracted summary model addresses is that the summary task's input text is lengthy, surpassing the pre-trained language model's maximum input length. In order to address this problem, the long text text summary problem is modeled as a sentence-by-sentence sequence annotation problem, and the model determines which sentences should be chosen as the abstracted summary. The task of abstracted summary model, is to abstract sentences containing important information as much as possible, and reduce the length of the original text as much as possible without losing important information. The general layout of the abstracted summary model is illustrated in Fig.2.

Fig.2 illustrates how the model can be split into two sections based on whether training fine-tuning is used. One is the BERT encoder part that freezes parameters and does not participate in fine-tuning, and the other is the convolutional neural network part that requires training fine-tuning. In the encoder part of obtaining article vectors, the whole article is taken as a raw input, the article is divided into several sentences using clause function, and each sentence is encoded using BERT model[29]. Then the sentence vector of the sentence is obtained through average pooling operation, and the vector representation of the original article is obtained by combining the sentence vectors of all the sentences of an article. This step can be done in advance and saved to a file. The convolutional network model of sentence abstraction is based on dilated convolution and gated convolution, and combined with fully connected network to classify and output sequence labeling problems.

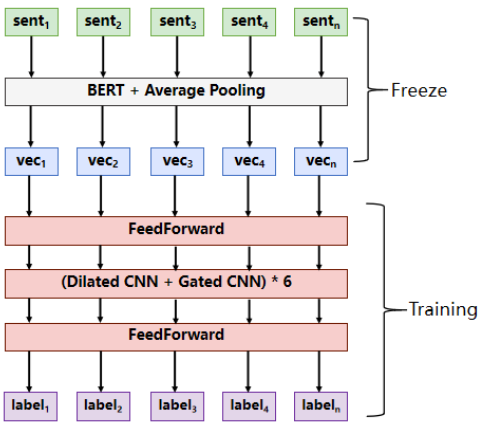

FIGURE 2. Schematic diagram of an overall abstracted summary model

The majority of the abstracted summary model is comprised of the following two components:

(1) Text coding part: The purpose is to encode an article in sentence units, and combine sentence vectors into article vectors as the input of the next part of the neural network model. This part mainly includes abstractive label conversion, embedding layer, Transformer layer and average pooling layer.

(2) Sentence abstraction: The classical TextCNN model is improved by taking the vector obtained from the BERT encoding part as input, and the focus is on the optimization and improvement of the convolution structure. It mainly includes increasing dilated rate parameters and using gated convolution.

1) ABSTRACTIVE LABEL CONVERSION

The output of the abstracted summary model corresponds to the original sentence in the article, whereas the annotated summary provided by the dataset is artificially generated, resulting in inconsistency. Therefore, this paper designs a label conversion algorithm to convert the manual summary into abstracted summary. The algorithm needs to make the abstracted sentences contain the information in the manual summary to the greatest extent. The algorithm rules are as follows:

(1) Create an algorithm for clauses to separate the source text and abstract into more finer-grained sentences;

(2) Go through each sentence in the manual summary, match the most relevant sentence in the original text, that is, the highest ROUGE score, delete the sentence that has been matched.

(3) Repeat (2) until the match is complete, and the selected sentences are de-duplicated and arranged in the order of the original text to serve as labels for the abstracted summary.

2) EMBEDDING LAYER AND AVERAGE POOLING LAYER

In this paper, BERT trained Embedding with dynamic coding is adopted. The three embedding vectors that make up the BERT model's input are stated as follows:

$$Input = Word_{emb} + Segment_{emb} + Position_{emb} \quad (1)$$

Where $Word_{emb}$ is the character embedding vector, $Segment_{emb}$ is the sentence embedding vector, and $Position_{emb}$ is the position embedding vector. The location coding of BERT models is randomly initialized before pre-training, and then learned during pre-training. The Embedding diagram of the embedding layer is shown in Fig.3.

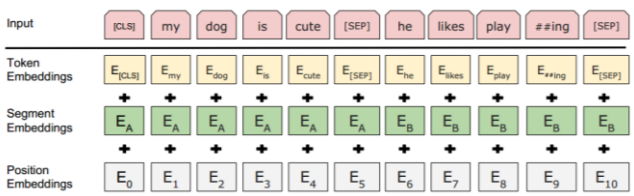

FIGURE 3. Bert Embedding diagram

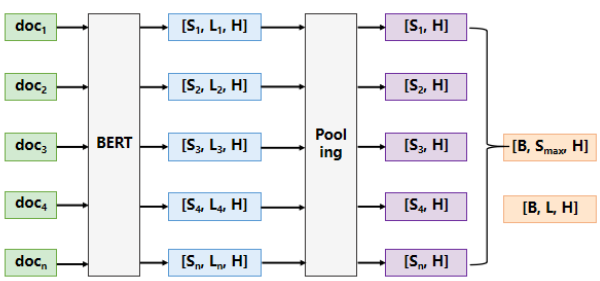

FIGURE 4. Dimension transformation of the text encoding part

The BERT encoder is utilized to encode each article, with the number of sentences being equivalent to the batch size, resulting in the generation of a three-dimensional vector for each article, then an article can be encoded into a three-dimensional vector, the dimensions of which are $[S_i, L_i, H]$, where S is the number of sentences, L is the article's maximum sentence length, and H is the hidden dimension. For different articles, the size of L is uncertain, so using the average pooling layer to average the sentence length is equivalent to smoothing out the difference between sentence lengths, and then the vector dimension of an article becomes $[S_i, H]$. In the later model, in order to conduct batch training, the padding operation is used to expand the dimension of sentence number S, then the dimension of a batch of articles becomes $[B, S_{max}, H]$, where S_{max} is equal to the maximum value of all S_i, which is replaced by a general symbol $[B, L, H]$. The

dimension transformation of the encoding part is shown in Fig.4.

3) DILATED CONVOLUTION

In this model, TextCNN model is used as the basic architecture, and on this basis, dilated rate parameters and gated structure are added to one-dimensional convolution. In TextCNN, the long-range dependence of a neuron inside a convolutional neural network is referred to as the size of the receptive field. Generally, the larger the convolutional kernel and the more layers stacked, the larger the receptive field will be. However, increasing the convolution kernel also means increasing the number of parameters and the amount of computation, and stacking more layers may cause the model to fail to update the gradient during training. Therefore, in order to enlarge the receptive field of neurons, dilated convolution is chosen in this model when the convolutional nuclei are small and the network layers are not deep. Fig.5 is a comparison diagram of standard convolution and dilated convolution.

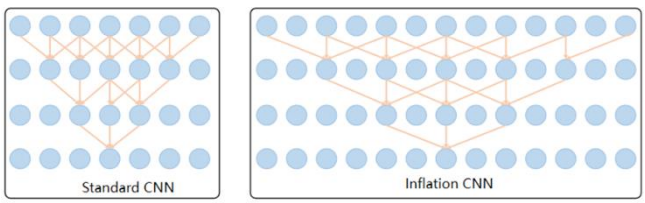

FIGURE 5. Standard convolution and dilated convolution

The two convolutional networks in the figure have convolution kernel sizes of 3, the dilated rate of the ordinary convolution on the left is 1, and the expansion rate of the dilated convolution on the right is [1,2,4]. It can be obtained by calculation that the receptive field of the ordinary convolutional network in the third layer is 7, which is less than 15 of the dilated convolutional network. The receptive field is enlarged, but the parameters remain the same, and the performance of the model is not affected.

4) GATED CONVOLUTION

This model also uses a gated convolutional network. If the sequence is expressed as $X = [x_1, x_2, \dots, x_n]$, then the convolutions with a gated mechanism can be expressed as:

$$Y = Conv1D_1(X) \otimes \sigma(Conv1D_2(X)) \quad (2)$$

Where, $Conv1D_1$ and $Conv1D_2$ have the same size, but their weights are different, so the number of arguments is twice that of ordinary convolution. Use the sigmoid function to activate one of the convolution variables as a gating unit, and the other convolution variable directly as a normal convolution output, and multiply the two variables bit by bit. Since the range of the activation function is (0,1), it is equivalent to adding a gate to each value of the output, which is used to control how much can be passed, which is similar to the improvement of the LSTM network on the RNN network. In addition, since the gate structure controls the flow, the risk of the gradient disappearing is also lower. The model also makes advantage of residual joins because the input and output have the same dimensions. The residual connection is combined with gated convolution as shown in formula 3, and formula 3 is deformed to obtain formula 4.

$$Y = X + Conv1D_1(X) \otimes \sigma\big(Conv1D_2(X)\big) \quad (3)$$

$$Y = X \otimes (1 - \sigma\big(Conv1D_2(X)\big)) + (X + Conv1D_1(X)) \otimes \sigma\big(Conv1D_2(X)\big) \quad (4)$$

Where $Conv1D_1(X)$ does not use the activation function, it is actually just a linear transformation of X, and adding X is also a linear change, so $X + Conv1D_1(X)$ is equivalent to $Conv1D_1(X)$, then formula 4 can be equivalent to:

$$Y = X \otimes (1 - \sigma) + Conv1D_1(X) \otimes \sigma \quad (5)$$

Where $\sigma = \sigma\big(Conv1D_2(X)\big)$. It can be clearly seen from the formula that information X passes through the convolutional network with a probability of $1 - \sigma$, and passes through the network with a gated convolutional calculation with a probability of σ. It can be seen that the residual connection can not only reduce the risk of gradient disappearance, but also propagate information in multiple channels, so the model is able to get more useful information features. The information flow in the gated convolutional network is shown in Fig.6.

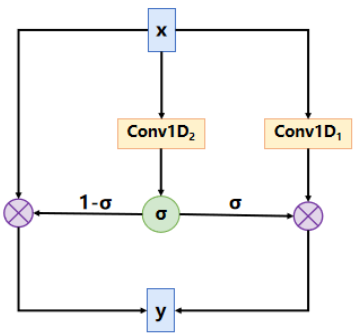

FIGURE 6. Information flow in a gated convolutional network

B. GENERATED SUMMARY MODEL

Fig. 7 displays the generated summary model's structure.By transforming the Attention matrix, it effectively integrates the bidirectional coding capability and the unidirectional text generated capability, so that the two-way language model has the text generated capability, and thus realizes the concise and elegant seq2seq structure. At the same time, NEZHA[30] model, which has significant advantages in dynamic encoding of Chinese text, is introduced as the embedding vector of text acquisition at the embedding layer. NEZHA replaces BERT's random mask with full word mask during the training process, which can make the character embedding vector contain rich context information and entity boundary information, and enhance the model's abstracted effect of text global semantics. A novel replication mechanism is designed to model the problem of whether to copy words or fragments from the original text as a sequence labeling problem, thus realizing a new replication mechanism. By introducing sparsity, the decoding method of cluster search is improved, and the learning effect of the model for key words is improved.

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2024.3427390

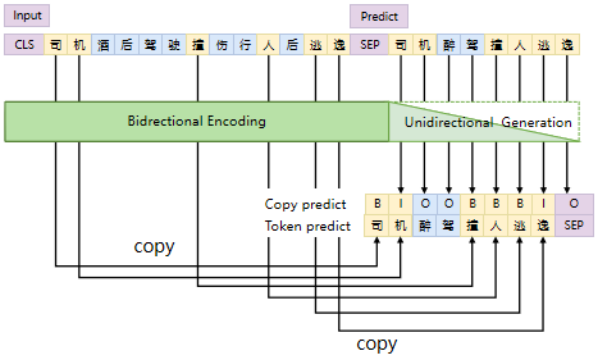

FIGURE 7. Schematic diagram of the generative summary model

1) TRANSFORMATION OF TWO-WAY LANGUAGE MODEL

A one-way language model is a text generated model that calculates the probability of the next word word-by-word in an autoregressive way, and can only calculate the current word based on the word that has already been generated. Taking "I love Peking University" as an example, this process is represented by the Attention Mask matrix, as shown in Fig.8.

FIGURE 8. Attention Mask matrix of one-way language model

The yellow block means that the token represented by this row is able to "notice" the token represented by this column, and the white block means that it is not able to "notice". The two-way language model can not only "notice" the above, but also "notice" the below. Take "I love Peking University" as an example again, which is also represented by the Attention Mask matrix, as shown in Fig.9.

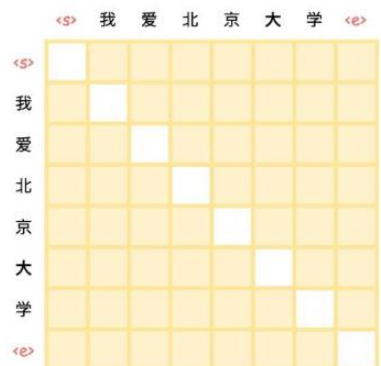

FIGURE 9. Attention Mask matrix of the two-way language model

The transformation of the two-way language model to make it have the text generated ability of the one-way language model is also the transformation of the Attention Mask matrix. The input part is a two-way language model, the output part is a one-way language model, and you need to make sure that the output part is "aware" of the entire input part. The transformed Attention Mask matrix is shown in Fig.10.

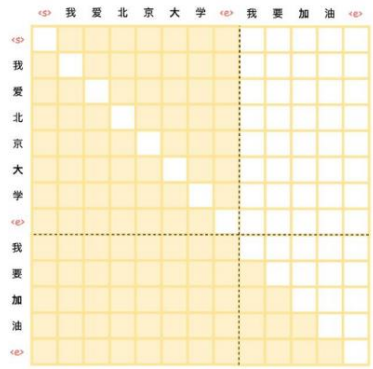

FIGURE 10. Example of the Attention Mask matrix of the seq2seq model for generating class tasks

2) REPLICATION MECHANISM

A replication mechanism is introduced to ensure the model focuses on key information without disregarding other text information. In order to solve the defect that the conventional replication mechanism does not consider the phrase of the word when copying a word, this paper uses a replication mechanism based on sequence labeling. By predicting the label of the current word, it can choose not to copy, copy the first word or copy the non-first word of the phrase, so that the ability of copying the phrase is greatly improved. The normal copying mechanism is to model the distribution of each word in the output section:

$$p(y_t|y_{<t}, x) \tag{6}$$

This model uses an additional sequence labeling task added to the output portion of the model as a replication mechanism, that is, predict one more label distribution z_t, and predict the new distribution:

$$p(y_t, z_t)|y_{<t}, x) = p(y_t|y_{<t}, x)\, p(z_t|y_{<t}, x) \tag{7}$$

Where $z_t \epsilon \{B, I, O\}$, B indicates a single word or the initial word of a word that has been copied verbatim from the source text; I indicates that the term being used is a copy of a word that is not the first word in the original text; O means that the current word belongs to the model generation rather than the copy. During the training stage, the algorithm directly obtains the B, I, O labels, and marks the public words and fragments that appear in the longest common subsequence of the input sequence and output sequence with copied labels. In the prediction phase, the prediction label z_t is first: if z_t is predicted to be O, then nothing is done and the output is decoded directly from the dictionary; If z_t predicts B, then only words that have appeared in the input text are considered; If z_t is predicted to be I, only the word that appears in the input text is considered, and the previous word of the word is also copied. According to the above process, the decoding process of the model is still step-by-step, but the unnecessary

words can be masked by the predicted label results, and only the possible words can be selected, so as to ensure that the needed parts are copied from the original text.

3) IMPROVED CLUSTER SEARCH

During the prediction generation stage, it is necessary to compute the likelihood of each word in the vocabulary being the output word at the current time, based on the preceding output word. Subsequently, one must calculate the probability distribution to determine the output probability of each word in the vocabulary at the next moment. However, as the vocabulary expands, the amount of computation will increase exponentially, and it is difficult to achieve NP-complete search. Therefore, scholars usually adopt a simpler algorithm to approximate the complete search with a local optimal solution, and if a given level of accuracy is met, accomplish the goal of cutting down on computing time and space. An enhanced cluster search technique is used in this paper.

The cluster search algorithm is a heuristic approach to graph exploration. As the clustering search algorithm iteratively grows its depth in a vast solution space, certain low-quality nodes will be pruned while higher-quality nodes are kept in order to minimize space consumption and increase time efficiency for the search. The best first search algorithm has been optimized in this way. In the best first search algorithm, all possible solutions are sorted according to heuristic rules to measure how close the obtained solution is to the target solution. Cluster search, on the other hand, uses a breadth-first search to build its search tree, cutting out the other nodes at each depth, keeping only the n most qualified solution nodes, and using a heuristic function to evaluate the power of each node it examines. However, in the calculation of the cluster search algorithm, there is a possibility that the potential best solution has been cut off at each time step. Fig.11 shows the algorithm diagram when the bunching width parameter is set to 2.

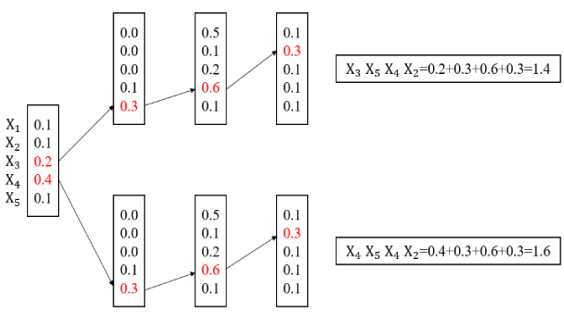

FIGURE 11. Algorithm flow structure when the bunching width is 2

The cluster search algorithm uses softmax function to normalize all the words in the dictionary when decoding the probability of the current word, which has high time complexity. Research based on softmax sparsity shows [31-32] that in problems with a large number of categories (such as dictionary size), sparsity processing of candidate categories will have better results. Due to the limited number of candidate words in the dictionary, only a relatively small portion is expected to be chosen. And most of the remaining numbers are not even in the range of candidates. Therefore, the model adopts sparse processing here, only retaining the 10 candidate words with the highest probability, and setting the probability of the remaining words to 0.

III. EXPERIMENTAL ANALYSIS

A. DATA SETS AND EVALUATION INDICATORS

The dataset utilized in this paper is the CNewSum dataset, which was introduced by Toutiao Platform in 2021, which is a large-scale Chinese news summary dataset containing 304,300 articles from Toutiao and human summaries provided by Toutiao Platform. The average length of the article's text is 730.4, while the manual abstract has an average length of 35.1. The dataset comprises 275,600 training sets, 14,400 verification sets, and 14,400 test sets. The lengthy documents contain highly abstract concepts that encourage a comprehensive understanding of the document and the creation of a current summary model. The test set of CNewSum is distinguished by its ample and inferential annotations of abstracts, providing a robust platform for automated Chinese abstract research.

The ROUGE method, proposed by Lin[33], is widely utilized for evaluating the performance of automatic abstracting models. The primary idea is to compare the summary produced by the model with the reference summary and then use a quantitative measure of the degree of basic unit overlap between the two sets of abstractions to determine the quality of the system summary. The evaluation indexes ROUGE-1, ROUGE-2, ROUGE-L, etc. are frequently used, where 1,2, and L represent based on 1-element word, 2-element word, and the longest string, respectively. This method is one of the general standards of abstract evaluation system, and the calculation formula is shown in formula 8:

$$R_{ROUGE-N} = \frac{\sum_{S\in\{Ref\}}\sum_{N_{n-gram}\in S} Count_{match}(N_{n-gram})}{\sum_{S\in\{Ref\}}\sum_{N_{n-gram}\in S} Count(N_{n-gram})} \tag{8}$$

Where n-gram means n-word, $\{Ref\}$ indicates the reference abstract, $Count_{match}(N_{n-gram})$ indicates the number of n-grams appearing in both the system abstract and the reference abstract, and $Count(N_{n-gram})$ indicates the number of n-grams appearing in the reference abstract.

B. EXPERIMENTAL ENVIRONMENT AND PARAMENTER CONFIGURATION

The operating system of this experiment is Linux, the processor is Inter Core i7, the GPU is GeForce RTX 3090, the program language is Python3.8.3, and the framework is Pytorch1.7.1. The model parameters are shown in Table I. The experimental loss curve is shown in Fig 12.

TABLE I MODEL PARAMETERS TABLE

Argument	Value	Implication
Batch_size	2	Batch size
Accumulation_steps	2	Cumulative gradient frequency
Learing_rate	2e-5	Learning rate
Warmup_proportion	0.1	Learning rate preheating ratio
Epochs	50	Number of iterations
Beam_size	5	Bunching width
Top_k	10	Sparse retention number
Optimizer	Adam	optimizer
Beta_1	0.9	Parameter β_1 of Adam
Beta_2	0.999	Parameter β_2 of Adam
Epsilon	1e-8	Parameter e of Adam

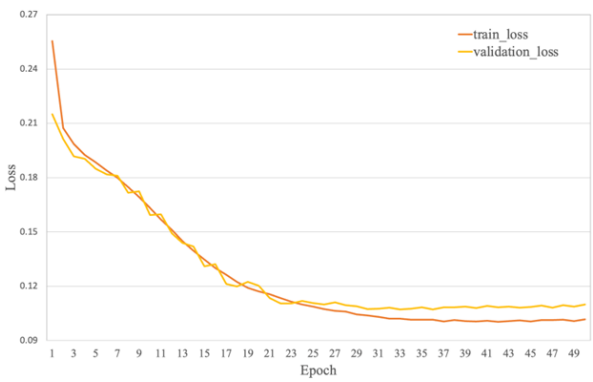

FIGURE 12. Training and test Loss diagram

C. ANALYSIS OF ABLATION EXPERIMENT

1) ABSTRACTED SUMMARY MODEL

Ablation tests were carried out to evaluate the influence of the BERT model, dilated convolution and gated convolution on the suggested abstracted model in this study. Table II displays the findings of the experiment.

TABLE II ABSTRACTED MODEL ABLATION EXPERIMENT

Models	Rouge-1	Rouge-2	Rouge-L
Ext-without-bert	26.24	14.32	21.62
Ext-without-dilation	33.74	19.57	28.56
Ext-without-gate	31.87	18.73	27.53
Ext（This paper abstracted model）	**34.39**	**20.95**	**29.37**

Table II illustrates how the three ablation experimental models' performance has declined to varying degrees in comparison to the abstracted model suggested in this paper. The dynamic coding BERT model helps the model to better abstract the original text information, and different expansion rates capture more scale contextual semantic information, making the model more flexible and effective. The dilated convolutional network can effectively enhance the model's understanding of the text context by improving the model's receptive field. The sigmoid function in gated convolution is used to calculate the value of the gated structure, which is similar to the effect of forgetting gate in RNN. The nonlinear activation value controls which information in the original convolution can be passed to the next layer, which not only adds nonlinear activation to the linear original convolution, but also effectively reduces the gradient dispersion in the training process and improves the performance of the original abstracted summary model.

2) GENERATED SUMMARY MODEL

In order to verify the improvement strategy of the generated model, the results obtained from the abstracted model were input into the generated model, and the following ablation experiments were designed for each module.

(1) In order to explore the impact of using NEZHA model as the embedding layer, the Chinese versions of BERT and RoBERTa models were compared and named bert_seq2seq and roberta_seq2seq respectively.

(2) In order to test the unidirectional generated summary model of fusion bidirectional coding by transforming the Attention matrix, a classic encoder-decoder structure model is adopted to conduct comparative experiments, and the model is named abs-use-transformer.

(3) In order to test the role of the copy mechanism in the model, the copy mechanism was removed from the original model and the model was named abs-without-copy.

(4) In order to test the function of sparse cluster search in the model, the sparse part was removed from the original model and the new model was named abs_without_sparse.

TABLE III RESULTS OF ABLATION EXPERIMENT

Models	Rouge-1	Rouge-2	Rouge-L
Bert_seq2seq	42.65	26.03	38.54
Roberta_seq2seq	44.36	26.94	38.73
Abs-use-transformer	41.76	26.22	38.86
Abs-without-copy	42.76	26.38	37.36
Abs-withoutt-sparse	43.34	26.74	38.12
Abs（This paper is a generated model）	**44.21**	**27.52**	**39.03**

Table III shows the results of the ablation experiment. Compared with the original model, the effects of the five models in the ablation experiment all decreased to varying degrees, and only the Rouge-1 value of the roberta_seq2seq model was slightly stronger than that of the original model. The experimental results show that, compared with BERT model and RoBERTa model, the NEZHA model is better than the NEZHA model in the Chinese context. Comparing the optimized seq2seq structure with the traditional encoder and decoder structure, the results show that the performance of the

unidirectional generated summary model is better than that of the unidirectional generated summary model, which integrates the two-way language model. In comparison experiments between Abs-withoutt-sparse model and ABS-Withoutt-sparse model, the effectiveness of the replication mechanism for copying a text fragment was verified, and the improved sparse cluster search also improved the learning ability of the model.

D. COMPARATIVE EXPERIMENTAL ANALYSIS

1) ABSTRACTED SUMMARY MODEL

In order to verify the performance of the abstracted model proposed in this paper, LEAD, TextRank, NeuSum, Transformer-ext and BertSum models are selected as comparison models to conduct comparison experiments. The comparison experiment results are shown in Table V.

TABLE V RESULTS OF COMPARISON MODEL EXPERIMENTS

Models	Rouge-1	Rouge-2	Rouge-L
LEAD	30.43	17.26	25.33
TextRank	24.04	13.70	20.08
NeuSum	30.61	17.36	25.66
Transformer-ext	32.87	18.85	27.59
BertSum	34.78	20.33	29.34
Ext（This paper abstracted model）	**34.39**	**20.95**	**29.37**

According to the experimental results, the abstract index of the abstracting abstract model proposed in this paper is better than that of LEAD, TextRank, NeuSum and Transformer-ext baseline models, and is on par with BertSum, the best baseline model. The scores of ROUGE-1 are slightly lower than that of BertSum model, while the scores of ROUGE-2 and ROUGE-L are slightly higher than that of BertSum model. It can be seen that the combination of different expansion rates and BERT model can better abstract the original text information, improve the model's sensitivity field to capture multi-scale context information, and increase the nonlinear of the model through the activation function, effectively reduce the gradient dispersion in the training process, and improve the performance of the abstracted summary model.

2) GENERATIVE SUMMARY MODEL

In order to verify the effect of the generated model on the text summary generation task, the results obtained from the abstracted model are input into the generated model, and the SumCoT, GEMINI, Pointer Generator, Transformer-abs, and BertSumAbs are selected as the benchmark models for comparison to verify the performance of the generated summary model in this paper. Table IV shows the experimental results of the generated summary model and the baseline model in this paper.

Table IV shows that the developed summary model in this research has more impacts than the five baseline models that were chosen, and its performance is slightly better than the best BertSumAbs model among the baseline models. This shows the effectiveness of the generated summary model on the common data set, which is due to the structure of the model and the comprehensive use of multiple mechanisms.

TABLE IV COMPARATIVE EXPERIMENTAL RESULTS

Models	Rouge-1	Rouge-2	Rogue-L
Pointer Generator	25.70	11.05	19.62
Transformer-abs	37.37	18.62	30.62
BertSumAbs	44.18	27.37	38.32
SumCoT[34]	43.88	27.01	37.91
GEMINI[35]	44.05	26.70	38.13
Abs（This generaed model）	**44.21**	**27.52**	**39.03**

IV. SUMMARY

The generation of high quality abstracts is challenging due to the problem of redundant information processing in long text. This research proposes a two-stage summary model that combines abstracted and generated summary models. Firstly, the sentences are encoded by BERT model, then the timing information between sentences is given to the gated convolutional network, and the long distance dependence of sentences is given to the dilated convolution network. In the generated summary model, a special Attention matrix is designed to transform the two-way language model, which effectively integrates the bidirectional coding ability and the unidirectional text generation ability, so that it can perform the text summary task. A new replication mechanism is designed. By the way of sequence annotation, the corresponding labels are predicted for the fragments that need to be copied, and the replication is decided according to the labels. Improved cluster search. The sparse processing of softmax function in cluster search avoids overlearning of the model and enables the model to focus on learning potential candidates, further improving the performance of the generated model.

In the future, there are still a lot of work that can be expanded, such as applying the generated algorithm to the generation of multi-document and multi-sentence summary, and how to abstract the key topic information of different granularity or different modes of global and local text more effectively.

REFERENCES

[1] Wu P, Zhou Q, Lei Z, et al. Template oriented text summarization via knowledge graph, 2018 International conference on audio, language and image processing (ICALIP). IEEE, 2018, pp.79-83.
[2] Huang L, Wu L, Wang L. Knowledge graph-augmented abstractive summarization with semantic-driven cloze reward[J]. arXiv preprint arXiv:2005.01159, 2020.
[3] El-Kassas W S, Salama C R, Rafea A A, et al, Automatic text summarization: A comprehensive survey. Expert systems with applications, 2021, 165: 113679.
[4] Luhn H P, "The automatic creation of literature abstracts," IBM Journal of research and development, vol. 2, no. 2, pp. 159–165, Apr. 1958.

[5] Radev D R, Jing H, Styś M, et al, "Centroid-based summarization of multiple documents," Information Processing & Management, vol. 40, no. 6, 2004, pp. 919–938.
[6] Wan X, Yang J, Xiao J, "Manifold-Ranking Based Topic-Focused Multi-Document Summarization," in International Joint Conference on Artifical Intelligence, Hyderabad, India,2007, pp. 2903-2908.
[7] Wen-Qian J, Zhoujun L, Wenhan C,et al, Automatic abstracting system based on improved lexRank algorithm, Computer Science, vol. 37, no. 5, 2010, pp. 151–154.
[8] Hirao T, Yoshida Y, Nishino M, et al, "Single-document summarization as a tree knapsack problem," in the 2013 conference on empirical methods in natural language processing, Washington, USA, 2013, pp. 1515-1520.
[9] Li S, Ouyang Y, Wang W, et al, Multi-document summarization using support vector regression. Proceedings of DUC, Citeseer, 2007, pp.42.
[10] Nishikawa H, Arita K, Tanaka K, et al, "Learning to Generate Coherent Summary with Discriminative Hidden Semi-Markov Model," in the 25th International Conference on Computational Linguistics, Ireland, 2014, pp.421648-1659.
[11] Rush A M, Chopra S, Weston J, "A neural attention model for abstractive sentence summarization," Computer Science, 2015. DOI:10.18653/v1/D15-1044.
[12] Bahdanau D, Cho K, Bengio Y, "Neural machine translation by jointly learning to align and translate," Computer Science, 2014.DOI:10.48550/arXiv.1409.0473.
[13] Chopra S, Auli M, Rush A M, "Abstractive Sentence Summarization with Attentive Recurrent Neural Networks,"in Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies ,San Diego, California 2016, pp. 93-98. DOI:10.18653/v1/N16-1012.
[14] Nallapati R, Zhou B, Gulcehre C, et al, "Abstractive text summarization using sequence-to-sequence rnns and beyond," 2016, arXiv preprint arXiv:1602.06023.
[15] Tan J, Wan X, Xiao J, From Neural Sentence Summarization to Headline Generation: A Coarse-to-Fine Approach, IJCAI. 2017, 17, pp. 4109-4115.
[16] Tan J, Wan X, Xiao J. Abstractive document summarization with a graph-based attentional neural model, Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). 2017, pp. 1171-1181.
[17] Hu B, Chen Q, Zhu F, "Lcsts: A large scale chinese short text summarization dataset," arXiv preprint arXiv:1506.05865, 2015. DOI:10.18653/v1/D15-1229.
[18] Gu J, Lu Z, Li H, et al, "Incorporating copying mechanism in sequence-to-sequence learning," arXiv preprint arXiv:1603.06393, 2016. DOI:10.18653/v1/P16-1154.
[19] See A, Liu P J, Manning C D, "Get to the point: Summarization with pointer-generator networks," arXiv preprint arXiv:1704.04368, 2017. DOI:10.18653/v1/P17-1099.
[20] Wang K, Quan X, Wang R, "BiSET: Bi-directional selective encoding with template for abstractive summarization," arXiv preprint arXiv:1906.05012, 2019. DOI:10.18653/v1/P19-1207.
[21] Babu G L A, Badugu S, "Deep learning based sequence to sequence model for abstractive telugu text summarization," Multimedia Tools and Applications, vol. 82, no. 11, pp. 17075–17096, 2023.
[22] Vo T, "A novel semantic-enhanced generative adversarial network for abstractive text summarization," Soft Computing, vol. 27, no. 10, pp. 6267–6280, 2023.
[23] Lewis M, Liu Y, Goyal N, et al, "Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension," in Meeting of the Association for Computational Linguistics，online, 2020.DOI:10.18653/V1/2020.ACL-MAIN.703.
[24] Raffel C, Shazeer N, Roberts A, et al, "Exploring the limits of transfer learning with a unified text-to-text transformer," The Journal of Machine Learning Research, 2020, 21(1): 5485-5551.vol. 21, no. 1, pp. 5485–5551, 2020.
[25] Liu Y, Ni A, Nan L, et al, "Leveraging locality in abstractive text summarization," arXiv preprint arXiv:2205.12476, 2022.
[26] Feijo D V, Moreira V P, "Improving abstractive summarization of legal rulings through textual entailment," Artificial intelligence and law, vol. 31, no. 1, pp. 91–113, 2023.
[27] La Quatra M, Cagliero L, "BART-IT: An Efficient Sequence-to-Sequence Model for Italian Text Summarization," Future Internet, vol. 15, no. 1, pp. 15, 2022.
[28] Zhao G B, Zhang Y B, Mao C L, et al, "Agenerative summary method of cross-border ethnic culture in corporating domain knowledge ," Journal of Nanjing University(Natural Sciences), vol. 59, no. 4, pp. 620-628, 2023.
[29] Devlin J, Chang M W, Lee K, et al, Bert: Pre-training of deep bidirectional transformers for language understanding, arXiv preprint arXiv:1810.04805, 2018.
[30] Wei J, Ren X, Li X, et al, "Nezha: Neural contextualized representation for chinese language understanding," arXiv preprint arXiv:1909.00204, 2019.DOI:10.48550/arXiv.1909.00204.
[31] Martins A, Astudillo R, From softmax to sparsemax: A sparse model of attention and multi-label classification, International conference on machine learning, PMLR, 2016, pp.1614-1623.
[32] Peters B, Niculae V, Martins A F T, "Sparse sequence-to-sequence models," arXiv preprint arXiv:1905. 05702, 2019. DOI:10.18653/v1/P19-1146.
[33] Lin C Y, "Rouge: A package for automatic evaluation of summaries,"In Proceedings of the Workshop on Text Summarization Branches Out (WAS 2004),2004, pp. 74-81.
[34] Wang Y, Zhang Z, Wang R. Element-aware summarization with large language models: Expert-aligned evaluation and chain-of-thought method, arXiv preprint arXiv:2305.13412, 2023.
[35] Bao G, Ou Z, Zhang Y. GEMINI: Controlling The Sentence-Level Summary Style in Abstractive Text Summarization, Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing. 2023: 831-842.

YANG CHEN was born in 1980 in Hefei, Anhui Province, China. She received her Bachelor's degree in Computer science and Applications from Hefei University of Technology in 1999 and her Master's degree in computer technology from Hefei University of Technology in 2012.

From 2003 to 2008, she was a teaching assistant at Anhui Xinhua University. From 2008 to 2018, she was a professional course lecturer at the School of Information Engineering, Anhui Xinhua University. Since 2018, she has been an associate professor at the School of Big Data and Artificial Intelligence at Anhui Xinhua University. Since 2020, he has been pursuing his PhD at the National University of the Philippines. She participated in the compilation of a provincial planning textbook, published more than 10 articles as the first author of various journal conference papers and a national invention patent. Her research interests include wireless network technology, natural language processing and more.

DR. ROBEN A. JUANATAS is a Professional Computer Engineer and an Associate Professor 3 at the National University Philippines, College of Computing and Information Technologies. He currently serves as the National and NCR Chapter President of the Institute of Computer Engineers of the Philippines, Inc. (ICpEP). Dr. Juanatas earned his Bachelor of Science in Computer Engineering from Adamson University, and his Master in Information Technology and Doctor of Technology from the Technological University of the Philippines. He was honored as the Most Productive Researcher and received the Most Cited Research Award in 2023.His research interests include Machine Learning, Neural Networks, Artificial Intelligence, and Engineering and Technology.