AN IMPROVED MT5 MODEL FOR CHINESE TEXT SUMMARY GENERATION

Fuping Ren[2], Jian Chen[1], and Defu Zhang[1]

[1]School of informatics, Xiamen University, Xiamen, 361005, China
[2]Shenzhen Comtech Technology Co. Ltd, Shenzhen 518063, China

ABSTRACT

Understanding complex policy documents can be challenging, highlighting the need for intelligent interpretation of Chinese policies. To enhance Chinese text summarization, this study utilized the mT5 model as the core framework and initial weights. Additionally, it reduced model size through parameter clipping, employed the Gap Sentence Generation (GSG) method as an unsupervised technique, and enhanced the Chinese tokenizer. After training on a meticulously processed 30GB Chinese training corpus, the study developed the enhanced mT5-GSG model. When fine-tuning on Chinese policy texts, it adopted the "Dropout Twice" approach and ingeniously merged the probability distribution of the two dropouts using the Wasserstein distance. Experimental results indicate that the proposed model achieved Rouge-1, Rouge-2, and Rouge-L scores of 56.13%, 45.76%, and 56.41% respectively on the Chinese policy text summarization dataset.

KEYWORDS

Natural Language Processing, Text Summarization, Transformer model

1. INTRODUCTION

The purpose of text summarization is to extract essential information from a given text or set of texts, commonly used for tasks like automatic report generation, news headline creation, and structured search previews. Text summarization methods are broadly categorized into Extractive Summarization and Abstractive Summarization. Abstractive Summarization can make full use of context information to achieve the coherence of summarization and conform to the thinking form of human natural language, but designing a good Abstractive Summarization method exists certain challenges.

Early Abstractive Summarization method was largely impractical. The introduction of the seq2seq framework[1] in 2014 garnered attention; however, it was plagued by issues such as generating inaccurate and duplicate information. Addressing these concerns, the Pointer-Generator Network (PGN)[2] proposed a hybrid pointer generation network to address word duplication and out-of-vocabulary words. Additionally, it employed a coverage mechanism to prevent the duplication of information. Most preceding text summarization models relied on RNN networks, leading to difficulties in parallelization.

The emergence of the Transformer model[3] in 2017 marked a significant milestone in the field of text summarization. However, conventional Transformer models did not exhibit dominance, paving the way for large-scale models to dominate both Extractive and Abstractive Summarization. The MASS model[4], introduced in 2019, addressed the limitations of the BERT

David C. Wyld et al. (Eds): IOTBS, NLTM, AIMLA, DBDM -2024
pp. 189-188, 2024. CS & IT - CSCP 2024 DOI: 10.5121/csit.2024.140214

model for generative tasks, proposing the use of continuous segments as masking objects and employing an entire Encoder-Decoder structure. Similarly, the BART model[5], also introduced in 2019, utilized an arbitrary noise function to perturb and reconstruct text within the Encoder-Decoder framework, making it more suitable for text summarization than previous methods. our model uses an abstractive method based on PEGASUS with a copy mechanism to generate the final summary from the bridging document. SUMOPE is proposed for long text summary generation, the results show that SUMOPE outperforms the state-of-the-art methods in terms of ROUGE scores and human evaluation.

Although most of the aforementioned models were designed for English, their application to Chinese text summarization remains challenging due to limited research and model availability. Large-scale models such as T5[8] and its multilingual variant mT5[9] have shown promising results for Chinese text summarization, albeit with time efficiency limitations. Google's PEGASUS model[10], proposed in 2020, specifically focused on sentence masking as an unsupervised task within an Encoder-Decoder framework, demonstrating excellent performance particularly on small sample datasets. However, PEGASUS faces limitations regarding parameter scale and representation ability, particularly in the context of Chinese summarization.

The contributions of this paper are as follows:
(1) Proposing an enhanced mT5 model based on GSG for Chinese Text Summary Generation, showcasing superior performance compared to other models.
(2) Introducing an improvement to the Dropout mechanism, resulting in enhanced performance through the execution of Dropout twice.
(3) Demonstrating the practical application of the proposed model for Chinese policy text summarization.

2. AN IMPROVED PRE-TRAINING MODEL MT5 BASED ON GSG AND MLM

This paper used the mT5 model as the basic framework and initial weight, then used the GSG method as an unsupervised task. This paper modified the bottom layer of the framework, cropped out a part of parameters to shrink the model, and improved the Chinese tokenizer. Finally, this paper used about 30G of the Chinese training corpus for training and obtained the mT5 pre-training model based on the GSG method.

2.1. Framework of the Proposed Model

The mT5 model is a versatile model that employs a unified "seq2seq" format to address various text-based NLP problems. Prior to pre-training, it is essential to consider the overall framework. mT5, based on the Transformer model, encompasses several transformer architectures, including Encoder-Decoder, Language Model, and prefix-based language models (Prefix LM). An experimental comparison of these three frameworks demonstrated that the Encoder-Decoder model yielded the best overall performance. Therefore, the mT5 model adopts the Encoder-Decoder framework in this paper.

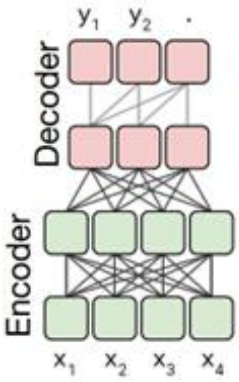

Figure 2.1 Encoder-Decoder architectures

Once the overall framework is established, the next step involves making policy choices for the training process. Various strategies are available for guiding the model, such as the Masked Language Model (MLM) method. For pre-training, the T5 model found that the BERT-like pre-training method was the most effective. This method primarily involves randomly destroying part of the content and then restoring it. Regarding the selection of the text destruction strategy, the T5 model identified the Replace spans method as the optimal choice, which considers entire words. The recommended text destruction ratio in the MLM model is generally 15%, and a subsection replacement length of 3 was found to be the most effective.

Based on these practical considerations, this paper ultimately adopts the Gap Sentence Generation (GSG) method. Leveraging the advantages and flexibility of the mT5 model, the model's basic framework and weights can be directly utilized.

2.2. Perfect Tokenizer

In addition, before training, the project needs to refine word segmentation. While the mT5 model and the Chinese BERT both use the sentencepiece tokenizer[11], some limitations were observed with its application in the Chinese context. Consequently, the paper decided to switch to the Tokenizer of BERT, incorporating improvements by adding the first 200,000 words from the Jieba word segmentation tool to the token dictionary of the original Chinese BERT. This modification aims to enhance lexical information for Chinese natural language processing models and improve vocab.txt by retaining only the highest frequency 100,000 terms and 50,000 characters.

2.3. Gsg method

GSG method requires three paragraphs of text, and masks the middle text. The method is shown in Figure 2.2. When using the GSG method in this paper, the MLM method is also added. There are a total of 3 sentences in Figure 2.2, the middle one "我住在厦门" is masked as Gap, marked as "[MASK1]". The surrounding text randomly selects words as the masking objects. In the Figure 2.2, the words "祖国" and "求学" were randomly selected and marked as "[MASK2]", and the proportion was still 15%. The Gap object is also input into the Decoder as the target text for text restoration. The masked way of the words is BERT-like, so the text restoration operation will be performed in the Encoder part.

The GSG method is proposed under the assumption that models closely aligned with downstream tasks can yield superior performance. It has demonstrated strong performance in text summarization tasks[9]. In fact, the GSG method is also a method of masking, but the object of masking is sentences. This aligns with the random substitution strategy of the T5 model, with the key distinction being the expansion of the small paragraph length into a full sentence. GSG offers three masking strategy options based on a given document $D = \{x_i\}_n$, where n represents the number of sentences and each sentence is denoted as x_i. The three strategies are as follows [9].

(1) Random: Randomly select m sentences as Gap Sentences.
(2) Lead: Select the previous m sentences as Gap Sentences.
(3) Principal: Select the previous m sentences as Gap Sentences according to the level of importance.
Among these options, Principal stands out as a relatively reasonable choice and is therefore adopted in this study. Two methods for assessing the importance of sentences are employed.
(1) Independent discrimination (Ind): The ROUGE1-F1 score is independently calculated for each sentence as an importance score for sorting, utilizing the calculation expression shown in formula (2.1).

$$s_i = rouge(x_i, D \backslash \{x_i\}), \forall_i \tag{2.1}$$

where, s_i represents the score of the i-th sentence, and the formula represents the relationship between the current sentence and the remaining text.

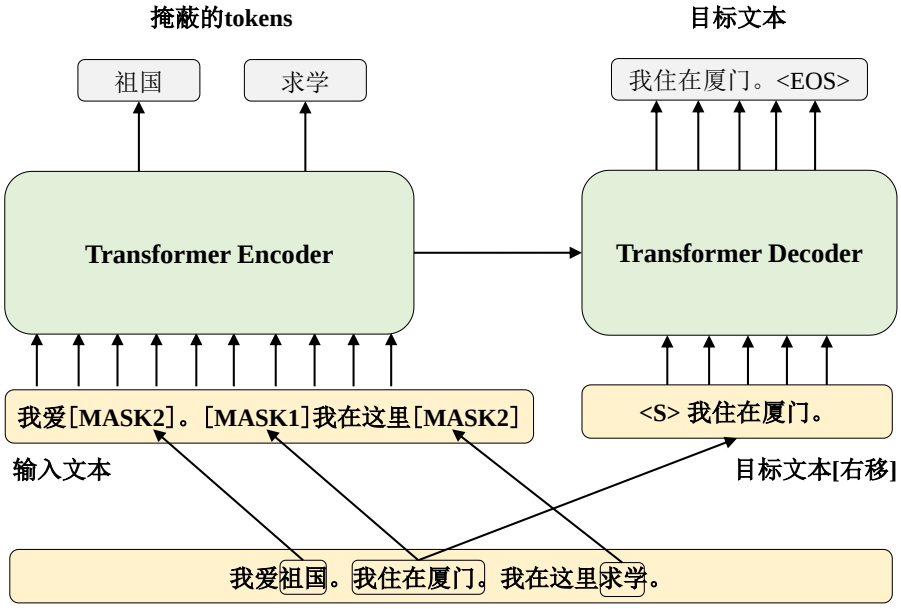

Figure 2.2 Schematic diagram of the GSG method

(2) Sequential discrimination (Seq): This method involves selecting the ROUGE1-F1 score of $S \cup \{x_i\}$ and the remaining text $D \backslash (S \cup \{x_i\})$ through Greedy Maximization, until m sentences are selected. The process is shown in Algorithm 2.1.

When calculating the ROUGE1-F1 score, n-gram (n-gram grammar) are categorized into two types: Non-repetitive n-gram set (Uniq) and Repeated n-gram set (Orig). Non-repetitive n-gram set (Uniq) first process the sentence set, remove the repeated n-gram, and then use ROUGE1-F1

for calculation. Repeated n-gram set (Orig) preserves the original sentence and allows n-gram repeated. This paper considered six combinations of the Principal method and n-gram, namely Ind-Uniq, Ind-Orig, Seq-Uniq, Seq-Orig, Random, and Lead. In this study, the Ind-Orig combination was selected. Additionally, the choice of gap sentences is proportional and referred to as Gap Ratio, with the most effective ratio identified as 30%.

Algorithm 2.1 Sequential Discrimination Algorithm

Algorithm 2.1 Selection of Gap Sentences for Sequential Discrimination

1: $S := \emptyset$

2: **for** $j \leftarrow 1$ to m **do**

3: $s_i := rouge\big(S \cup \{x_i\}, D \backslash (S \cup \{x_i\})\big), \quad \forall_i \ s.t. \ \ x_i \notin S$

4: $k := argmax_i\{s_i\}_n$

5: $S := S \cup \{x_k\}$

6: **end for**

In summary, this paper chose the mT5 model as the initial weight and fundamental framework, adhering to the standard Encoder-Decoder structure. Furthermore, for pre-training, a BERT-like method was employed. Regarding the masking strategy, a small segment mask (Replace spans) was utilized, while GSG method slightly differs by masking sentences. Considering the issue of masking ratio, a 30% Gap Ratio for the GSG method yielded the best results. Given that it involves the masking of sentences, the length of the span was no longer taken into account. For simplicity, the proposed model is subsequently referred to as mT5-GSG.

2.4. An Improved Dropout

In 2021, a simple improvement to Dropout known as "Dropout Twice" was proposed in SimCSE[12]. This enhancement involves executing Dropout twice to enhance its effectiveness. The rationale behind this approach lies in addressing the inconsistency problem between the training and inference stages resulting from the inherent randomness introduced by Dropout itself. To implement this improvement, consider a dataset $D = \{(x_i, y_i)\}_{i=1}^{n}$, the purpose of training is to obtain a model $P^w(y|x)$, which n represents the number of training samples, (x_i, y_i) represents a labeled sample pair, x_i represents the input data, and y_i is the label. Using "Dropout Twice" yields two distribution models P_1 and P_2, which can be combined using similarity metrics such as KL divergence [13], JS divergence [14], and Wasserstein distance adopted in this paper.

The Wasserstein distance, also known as earthmover's distance, measures the dissimilarity between two probability distributions and is given by the formula (2.2).

$$D_{ws}(P_1, P_2) = inf_{\theta \sim \Pi(P_1, P_2)} \mathbb{E}_{(x,y)\sim\theta}[||x - y||] \tag{2.2}$$

Here, inf refers to the largest lower bound, $\Pi(P_1, P_2)$ is a set of all possible joint distributions combining with the P_1 and P_2 distribution, and $\mathbb{E}_{(x,y)\sim\theta}[||x - y||]$ computes the distance between two samples *x* and *y* sampled from the joint distribution θ. Therefore, the expectation of

this sample pair distances under the joint distribution θ can be calculated. and the lowest attainable bound on this expectation across all possible joint distributions is the Wasserstein distance.

Based on the above, this paper utilized the Wasserstein distance to combine the distributions obtained from the two Dropouts. Prior to this, it is crucial to clarify that the primary aim of model training is to minimize the negative log-likelihood loss function, as expressed in the following formula (2.3).

$$L_{nll} = \frac{1}{n}\sum_{i=1}^{n} -logP^{w}(y_i|x_i) \tag{2.3}$$

For "Dropout Twice", the sample x_i is repeatedly input into the feedforward neural network, and will obtain two distributions, denoted as $P_1^w(y_i|x_i)$ and $P_2^w(y_i|x_i)$. For the same input (x_i, y_i), two unequal probability distributions are obtained. After two Dropouts, the negative log-likelihood function is shown in formula (2.4).

$$L_{nll}^{i} = -logP_1^w(y_i|x_i) - logP_2^w(y_i|x_i) \tag{2.4}$$

Considering the Wasserstein distance between the two Dropout distributions, we arrive at the formula (2.5).

$$L_{ws}^{i} = D_{ws}\left(P_1^w(y_i|x_i), P_2^w(y_i|x_i)\right) \tag{2.5}$$

Following the computation of the aforementioned formulas (2.4) and (2.5), values are derived using the negative log-likelihood function and the Wasserstein distance. To mitigate the influence of the Dropout module, an enhancement to the previous loss function is analogized by introducing influencing factors for adjustment. The final model incorporates the Wasserstein distance, as depicted in formula (2.6).

$$\begin{aligned} L^i &= L_{nll}^{i} + \beta L_{ws}^{i} \\ &= -logP_1^w(y_i|x_i) - logP_2^w(y_i|x_i) + \beta D_{ws}\left(P_1^w(y_i|x_i), P_2^w(y_i|x_i)\right) \end{aligned} \tag{2.6}$$

3. EXPERIMENTAL ANALYSIS

3.1. Evaluation Indicators & Data

The most widely used evaluation method in the text summarization domain is the ROUGE[15] evaluation metric, which commonly includes $ROUGE-N$ and $ROUGE-L$. These metrics can be computed as follows:

$$ROUGE-N = \frac{\sum_{S\in\{ReferenceSummaries\}}\sum_{gram_n\in S} Count_{match}(gram_n)}{\sum_{S\in\{ReferenceSummaries\}}\sum_{gram_n\in S} Count(gram_n)} \tag{3.1}$$

In the above equation, n denotes $n-gram$, $Count(gram_n)$ denotes the number of occurrences of one $n-gram$, and $Count_{match}(gram_n)$ denotes the number of co-

occurrences of one $n-gram$. Typically, the *N* values commonly range from 1 to 4, with this paper selecting 1 and 2.

$$ROUGE-L=\frac{(1+\beta^2)Rec_{lcs}Pre_{lcs}}{Rec_{lcs}+\beta^2 Pre_{lcs}} \tag{3.2}$$

$$Rec_{lcs}=\frac{LCS(X,Y)}{m} \tag{3.3}$$

$$Pre_{lcs}=\frac{LCS(X,Y)}{n} \tag{3.4}$$

Where X represents the candidate abstract, Y represents the reference abstract, $LCS(X,Y)$ represents the length of the longest common subsequence of X and Y. and *m* and *n* represent the lengths of *Y* and *X* respectively, Rec_{lcs} represents the recall rate, and Pre_{lcs} represents the precision rate. β is an influence factor, typically set to a large value.

Furthermore, in text summarization or text generation tasks, the decoder module usually employs a search algorithm during decoding. Commonly used methods include Greedy search and Beam Search [16], with this paper utilizing Beam Search.

Table 3.1 shows the experimental parameter settings during fine-tuning.

Parameter	Value
BERT hidden layer dimension	768
Learning rate when fine-tuning mT5-GSG	1e-5
Batch Size during mT5-GSG training	16
EPOCH	100
STEPS	500K
Optimizer	AdamW

Regarding datasets, this paper considers public Chinese text abstract datasets, including CSL, and NLPCC2017. In particular, a Chinese policy text abstract from a practical project is considered. The specific sample size is shown in Table 3.2 below.

Table 3.2 The size of the dataset samples (unit: pieces)

Data set	Train sample	Dev sample	Test sample
CSL	50000	500	200
NLPCC2017	50000	800	200
Project dataset	8000	100	50

3.2. Effects of Mt5-GSG

In this section, all models do not improve Dropout. Additionally, the proposed mT5-GSG model utilizes the Ind-Orig strategy with a Gap Ratio of 30%.

(1) The effect of different models on the CSL dataset

Table 3.3 shows the effect of different models on the CSL dataset. BERT-PGN [17], mT5, and PEGASUS [10] models were selected for comparison because they belong to the state-of-the-art models for Chinese text summary generation. Notably, the beam size significantly influences the models' performance.

The proposed mT5-GSG obtained the best results when the beam size is set to 3. The Rouge-1, Rouge-2 and Rouge-L scores are 70.45%, 60.57% and 68.26 %, respectively. Compared with the mT5 model, the Rouge-1, Rouge-2 and Rouge-L scores of mT5-GSG model are improved by 1.64%, 1.90% and 2.43% respectively.

Table 3.3 Comparison results of the models on the CSL dataset (unit: %)

Model	Beam Size	Rouge -1	Rouge -2	Rouge- L
BERT-PGN (Multidimensional Semantic Features)	2	42.70	16.64	38.44
PEGASUS	2	65.45	54.91	63.81
mT5	2	68.22	57.83	66.38
mT5-GSG	2	69.00	58.74	66.96
BERT-PGN (Multidimensional Semantic Features)	3	44.01	25.73	43.79
PEGASUS	3	66.34	56.06	64.75
mT5	3	68.81	58.67	66.83
mT5-GSG	**3**	**70.45**	**60.57**	**68.26**
BERT-PGN (Multidimensional Semantic Features)	4	43.87	17.50	38.97
PEGASUS	4	66.09	55.75	64.44
mT5	4	68.68	58.50	66.65
mT5-GSG	4	69.19	59.10	67.25

(2) The effect of different models on the NLPCC2017 dataset

Table 3.4 shows the experimental results of the models on the NLPCC2017 dataset. For mT5-GSG, the best performance was attained when the beam size equaled 3, resulting in Rouge-1, Rouge-2, and Rouge-L scores of 48.89%, 35.63%, and 43.04% respectively.

Table 3.4 Comparison results of the models on the NLPCC2017 dataset (unit: %)

Model	Beam Size	Rouge-1	Rouge-2	Rouge-L
BERT-PGN (Multidimensional Semantic Features)	2	41.12	23.55	34.46
PEGASUS	2	47.21	24.56	39.25
mT5	2	47.52	33.51	41.33
mT5 -GSG	2	48.67	33.39	42.07
BERT-PGN (Multidimensional Semantic Features)	3	42.28	23.89	35.63
PEGASUS	3	47.74	25.59	40.82
mT5	3	47.94	34.55	42.73
mT5-GSG	3	**48.89**	**35.63**	**43.04**
BERT-PGN (Multidimensional Semantic Features)	4	41.86	23.62	34.58
PEGASUS	4	47.68	25.27	40.54
mT5	4	47.83	34.47	42.49
mT5-GSG	4	48.78	34.90	42.91

(3) The effect of different models on the Chinese policy text summary dataset

Similarly, Table 3.5 presents the effect comparison of the models on the Chinese policy text summary dataset. Once again, mT5-GSG excelled notably when employing a beam size of 3, achieving Rouge-1, Rouge-2, and Rouge-L scores of 54.63%, 44.18%, and 55.24% respectively.

Table 3.5 Comparison results of the models on the Chinese policy text summary dataset (unit: %)

Model	Beam Size	Rouge -1	Rouge -2	Rouge- L
BERT-PGN (Multidimensional Semantic Features)	2	35.98	17.76	33.63
PEGASUS	2	50.77	35.59	50.95
mT5	2	48.25	21.35	36.69
mT5-GSG	2	53.01	28.27	54.91
BERT-PGN (Multidimensional Semantic Features)	3	36.15	17.54	33.63
PEGASUS	3	52.27	37.98	53.44
mT5	3	50.27	20.15	50.57
mT5-GSG	3	**54.63**	**44.18**	**55.24**
BERT-PGN (Multidimensional Semantic Features)	4	35.47	17.27	33.52
PEGASUS	4	51.91	37.09	50.38
mT5	4	50.03	26.23	49.52
mT5-GSG	4	53.74	41.40	54.85

4. CONCLUSIONS

This paper introduces a specialized pre-training model mT5-GSG, which utilizes the Gap Sentence Generation (GSG) approach for unsupervised training by integrating the framework and initial weights of mT5. Subsequently, model cropping is employed to reduce the model size, followed by pre-training on a Chinese corpus of approximately 30GB. Ultimately, an mT5-GSG pre-training model of about 370 million parameters is obtained, effectively resolving the challenges encountered by other models. To further enhance the model's performance, this paper proposes the "Dropout Twice" concept, which innovatively combines the probability distributions of two Dropouts using the Wasserstein distance method. The computational results demonstrate that this model outperforms existing models, particularly exhibiting optimal performance on Chinese policy text datasets. The Rouge-1, Rouge-2, and Rouge-L scores are 56.13%, 45.76%, and 56.41% respectively, satisfying the requirements of practical applications.

ACKNOWLEDGEMENTS

This work was jointly supported by the grant from the National Natural Science Foundation of China (61672439) and by the project grant from the Coding (xiamen) big data science company.

REFERENCES

[1] Sutskever I, Vinyals O, Le Q V. Sequence to sequence learning with neural networks. Advances in neural information processing systems, 2014, 27.
[2] See A, Liu P J, Manning C D. Get To The Point: Summarization with Pointer-Generator Networks.Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). 2017: 1073-1083.
[3] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need.Proceedings of the 31th International Conference on Neural Information Processing System. 2017: 6000-6010.
[4] Song K, Tan X, Qin T, et al. MASS: Masked Sequence to Sequence Pre-training for Language Generation//International Conference on Machine Learning. PMLR, 2019: 5926-5936.
[5] Lewis M, Liu Y, Goyal N, et al. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension//Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. 2020: 7871-7880.
[6] Zhao Y, Huang S, Zhou D,et al. CNsum: Automatic Summarization for Chinese News Text. Lecture Notes in Computer Science, 2022, 13472:539-547.
[7] Chang C, Zhou J, Zeng X, Tang Y. SUMOPE: Enhanced Hierarchical Summarization Model for Long Texts. Lecture Notes in Computer Science, 2023,14177:307-319.
[8] Raffel C, Shazeer N, Roberts A, et al. Exploring the limits of transfer learning with a unified text-to-text transformer. Journal of Machine Learning Research, 2020, 21: 1-67.
[9] Xue L, Constant N, Roberts A, et al. mT5: A massively multilingual pre-trained text-to-text transformer. arXiv preprint arXiv:2010.11934, 2020.
[10] Zhang J, Zhao Y, Saleh M, et al. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. International Conference on Machine Learning. PMLR, 2020: 11328-11339.
[11] Kudo T, Richardson J. SentencePiece: A simple and language independent subword tokenizer and detokenizer for Neural Text Processing.Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations. 2018: 66-71.
[12] Gao T, Yao X, Chen D. Simcse: Simple contrastive learning of sentence embeddings. arXiv preprint arXiv:2104.08821, 2021.
[13] Barz B, Rodner E, Garcia Y G, et al. Detecting regions of maximal divergence for spatio-temporal anomaly detection. IEEE transactions on pattern analysis and machine intelligence, 2018, 41(5): 1088-1101.
[14] Sra S. Metrics induced by Jensen-Shannon and related divergences on positive definite matrices. Linear Algebra and its Applications, 2021, 616: 125-138.

[15] Lin C Y. Rouge: A package for automatic evaluation of summaries. Text summarization branches out. 2004: 74-81.
[16] Zhao T, Ge Z, Hu H, et al. Generating Natural Language Adversarial Examples through An Improved Beam Search Algorithm. arXiv preprint arXiv:2110.08036, 2021.
[17] Jinyuan Tan, Yufeng Diao, Ruihua Qi, Hongfei Lin. Chinese News Text Automatic Abstract Generation Based on BERT-PGN Model. Journal of Computer Applications, 2021, 41(01):127-132.