Proceedings of the 3rd International Conference on Signal Processing and Machine Learning
DOI: 10.54254/2755-2721/5/20230543

Survey on abstractive text summarization using pretraining models and their developments

Yixin Zhang

China, 3003 Yipinyaju, Luoyang, Henan
United Kingdom, F167, the quadrangle, 1 Lower Ormond Street, M1 5QF
University of Manchester

yixin.zhang-15@student.manchester.ac.uk

Abstract. In these years, pre-training models gain a lot of attention in the summary generation area and demonstrate new possibilities for improving the sequence-to-sequence attention framework. This survey conducts a comprehensive overview of BERT-based pre-training models that can be used in abstractive summaries. Firstly, the BERT model is introduced as a typical pre-training model, followed by baseline models inspired by it. Then problems and developments of previous models are discussed including some recent SOTA approaches. Apart from that, some datasets used for models are demonstrated with main features. Besides, the commonly used evaluation methods are introduced. Last but not least, several potential research directions are suggested.

Keywords: pre-training model, abstractive summary, natural language processing, BERT, transformer

1. Introduction

Natural language processing, well known as NLP, which can be divided into Natural Language Understanding and Natural Language Generation subfields, was first introduced in the 1950s with the famous tuning test as a kind of rule-based method. In the 1970s, with the help of the high-speed development of the internet which enlarges the corpus and enhanced the hardware, statistics-based methods replaced rule-based [1]. It tackled the problem of the impossibility of using rules to cover all the linguistics that were naturally generated by rule-based methods [1]. Then comes the deep learning era. Following the traditional Recurrent Neural Network [2], LSTM [3], and GRU [4] tackled the gradient exploding and vanishing problem and performed well in a lot of NLP tasks. The word2vec model [5] was introduced by google research and strongly enhanced the performance of NLP at that age. In recent years, various forms of attention mechanisms that are based on Bahdanau attention [6] improve highly on the performance of traditional sequence-to-sequence models and pre-training models like GPT for Generative Pretrained Transformers [7] resolved the insufficient training data problem by using unsupervised pre-training to gain understandings of natural language and the corpus, and supervised fine-tuning method and became one of the most reliable models in NLP area.

Text summarization is one of the original tuning test tasks, which attempts to produce summaries which is concise, fluent, and most importantly, retain all the essential information in the original

DOI: 10.54254/2755-2721/5/20230543

document [1], which consists of extractive summaries and abstractive summaries, the former aims to select important sentences in the document to create a summary, while the latter one generates new sentences to conclude documents which means it includes both NLU and NLG tasks [1]. After the proposal of the attention mechanism, abstractive summary models focus on attaching various kinds of attention like attention inside the original text, attention inside summaries, attention between the original text and summaries, etc.

Recently, after the proposal of transformers [8], the development of abstractive summary mostly uses transformers as structures with mostly bidirectional pre-training methods, which means, inspired by GPT and Bidirectional Encoder Representation using Transformer (BERT, which is more focused on this survey) [9]. Based on that, some baseline models use new methods for pre-training like masking consecutively for MASS model and simulating noise of original document in pre-training for BART model, others use large language model like T5 to enhance performance in language generation or combining concepts from other deep learning areas like reinforcement learning and meta-learning. Apart from that, exploiting more information from the training set, and combining encoder and decoder can also enhance the performance of models. More attention was focused on creating multiple summaries and using labels to rank them for the most up-to-date research. In this survey, models will be grouped by performance promotion based on prior models. The main part can be divided into five parts, the first three talks about models the fourth introduces the dataset in the abstractive summary and the last part talks about some optimization methods that researchers proposed that can be used in the abstractive summary area and enhance the summary quality. In the end, some possible research directions in the abstractive summary area will also be provided to inspire readers in their generation's research.

2. Main part

This section discusses models, datasets, and optimization approaches in the abstractive summary area that used pre-training models. The first three sections discuss models which start with demonstrating the basic ideas of BERT model, the most famous transformer-based pre-training model, followed by several baseline models based on them, which contribute to the main achievements around 2019, then comes to the recent research achievements that mainly based on baseline models in the second part, the fourth part provides some brief introductions on datasets that used in models in previous parts, while the last part describes evaluation metric used for abstractive summary.

2.1. BERT and GPT

Proposed in 2018, Bidirectional Encoder Representation from Transformers, for short, BERT [9], uses bidirectional transformers as encoders and bidirectional pre-training which means for each masked token, all unmasked tokens can be used to predict it. Before training, the source text needs to be tokenized and embedded with token embedding for word vectors which add starting token [CLS] and sentence separate token [SEP], segment embedding to distinguish sentences in the pre-training procedure, and position embedding for positions inside sentences. The training of this model consists of pre-training and another training which is supervised and aims to fit downstream tasks, fine-tuning. After embedding, the model does a pre-training to obtain a basic understanding of the corpus, which consists of a masked bidirectional language model to predict masked tokens using all unmasked tokens, and a next sentence prediction to classify whether the second one is next to the first one and output in [CLS] token. Finally, for fine-tuning, BERT uses the weighted state of [CLS] token and SoftMax function for sequence level label probability:

$$P = softmax(CW^T) \tag{1}$$

Based on this fine-tuning function, most of the tasks can be solved by only adjusting the inputs and outputs of the model. For example, for question answering tasks like SQALD, use the question to be the first input sequence and the paragraph as the second sequence and predicting label 'start', 'end', and 'span' as the start, end, and the middle of answer for each token in the second sequence as output.

DOI: 10.54254/2755-2721/5/20230543

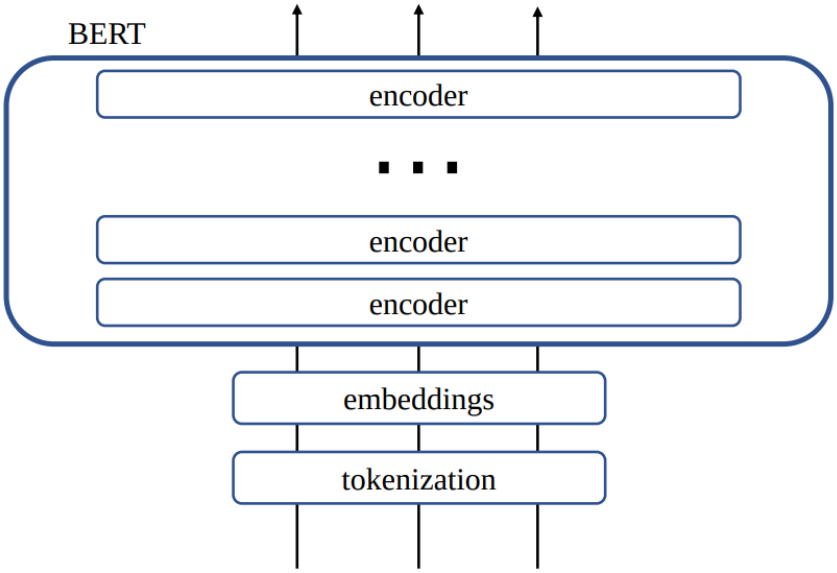

Figure 1. Structure of BERT.

By understanding context, BERT can be used in many downstream tasks, and by using transformers, BERT is more efficient and capable of longer-distance dependence compared with traditional RNNs. However, BERT model also has some problems, the most explicit one is the inconsistency between training and predicting as while predicting, masks used in training do not appear, and the model needs to auto-regressive the new summary which contributes to an error accumulation. The problem is also known as exposure bias and developments based on it is one of the main trends in recent years. Solutions to that problem are discussed in section 3 of the main part grouping by methods. Apart from that, BERT converges speed is slower compared to traditional left-to-right models as it predicts about 15% tokens rather than all, however, solutions to this problem are not included in this article.

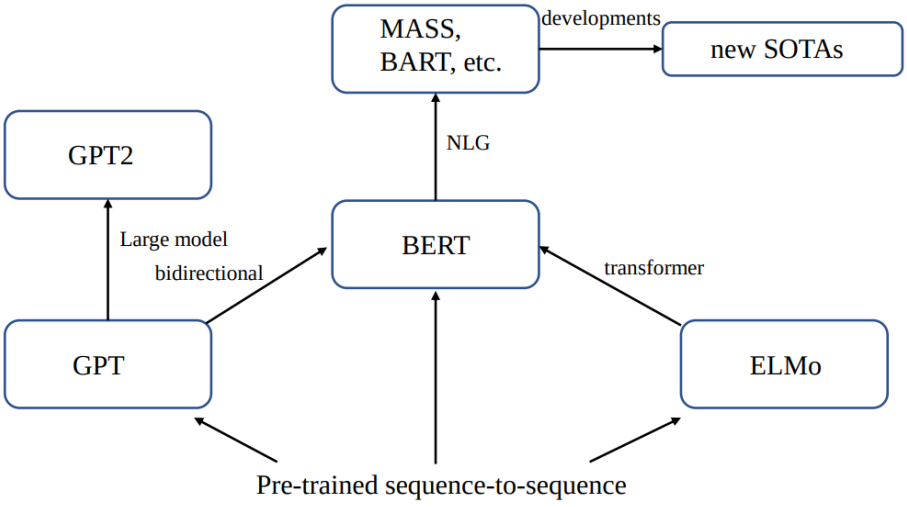

Figure 2. Relationships between models.

2.2. Based on BERT and sequence to sequence

Based on BERT pre-training, there are many ways to modify and create a generation model, for example, combines with the classical approach of text generation sequence-to-sequence model. In this part, pre-training models that use multi-layer transformers and pre-training, especially bidirectional ones, which means the idea of BERT, will be discussed and grouped by the main modification. The first section

DOI: 10.54254/2755-2721/5/20230543

focuses on the pre-training method as the main development, the second part discusses connecting sequence-to-sequence structures with BERT and large language models, and the third part introduces fine-tuning method. Models in this section are mostly qualified for both NLU and NLG except pointed out.

The MASS model uses encoder-decoder structures like the sequence-to-sequence model and pre-trains with several consecutive masked tokens instead of single ones in BERT. Differed from BERT which only pre-trains on the encoder, MASS pre-trains jointly on both encoder and decoder with encoder to predict the consecutively masked segments at first, then the unmasked parts in the source will be masked to force the decoder to rely more on source representation rather than previous tokens [10]. Another model, UNiLM combines sequence to sequence based on BERT model, is also a multi-layer transformer encoder network, which uses all single directional, bidirectional, and sequence-to-sequence language models for pre-training with shared parameters to create novel performance on both NLU and NLG [11]. However, UNiLM model only allows input lengths shorter than 512 tokens which is an explicit drawback for predicting long documents, and as it predicts word by word, so the inference would be relatively time-consuming. Based on it, the second version, UNiLMv2 uses pseudo-masks to insert [P] token on positions to be masked by using the same position embeddings for both tokens that should be masked and their masks, rather than the vanilla Masked Language Model which uses masks to replace the original token to enable both auto-encoder and partially auto-regressive on masked tokens [12]. Another widely used baseline model for abstractive summary, Bidirectional and Auto-regressive Transformers, a.k.a. BART, uses the idea of adding noise in the original text for pre-training by several methods: token masking to predict the content of token, token deletion which predicts both content and position of tokens, text infilling based on spanBERT to use consecutive masked tokens with Poisson distribution, sentence permutation which shuffles sentences in source documents and document rotation for predicting the starting sentences of documents. For structure, BART uses a sequence-to-sequence encoder-decoder structure but deleted the feedforward network and for the decoder added cross-attention [13]. In another form, using a pure sequence-to-sequence model but changing the predicting method, a new model named ProphetNet uses future n-gram prediction instead of only one gram with n-stream self-attention including the original masked multi-head self-attention and n self-attention stream for predicting on the future n-grams [14]. Moreover, designed specifically for abstractive summary, the PEGASUS model adds the gap sentence generation task to the original masked language model, which is to mask the important sentences measured by the Rouge1-F1 score and use other sentences to predict them [15]. However, the sentence selection strategy of PEGASUS only focuses on the importance of sentences but ignored factuality, to tackle that, FACTPEGASUS model considered factuality when selecting sentences, uses three complementary components for fine-tuning to remove hallucinations, improve factuality, make fine-tuning closer to pre-training, and added FACTCC to evaluate factuality on normal ROUGE score evaluation [16].

On the other hand, improvements can also be done by connecting structures and improving the model scale, and connecting different models. Connecting sequence to sequence and BERT, another method for NLG uses both encoder and decoder as a basic structure with an encoder for feature extraction and a decoder with the first part as a transformer to generate draft output and mask for the use of another BERT model to generate output summary [17]. Apart from that, the T5 model also has a great performance on natural language generation by using a large language model [18].

Besides improving on pre-training, structure, and scale, using a new method for fine-tuning can also make BERT fit for abstractive summary, BERTSUMAbs [20] uses BERTsum [19] as encoder and a randomly initialized 6-layer transformer as a decoder with a new two-staged fine-tuning schedule and not using the same optimizer for encoder and decoder to minimizing the inconsistencies between two segment embeddings.

DOI: 10.54254/2755-2721/5/20230543

Table 1. ROUGE-1, 2, L scores for models in section2 on CNN/DM dataset.

	R-1	R-2	R-L
MASS [10]	42.12	19.50	39.01
UNiLM [11]	43.33	20.21	40.51
UNiLMv2 [12]	43.16	20.42	40.14
BART [13]	44.16	21.28	40.90
Prophetnet [16]	43.68	20.64	40.72
PEGASUS [14]	44.17	21.47	41.11
no name [17]	41.71	19.49	38.79
BERTSUMAbs [20]	41.72	19.39	38.76
T5 [18]	43.52	21.55	40.69
FACTPEGASUS [15]	-	-	-

2.3. Problems and developments

Although the models above have great performance in abstractive summary, there are still some problems, especially those problems inherited from sequence-to-sequence ones. In this section, problems of the classical sequence-to-sequence models are discussed at first, followed by developments based on the models above, including pre-training and inputs developments, structure developments using models from the NLP area, combining models from other NLP areas, objective function developments, and using the ranking method for improvement.

The two main problems of sequence-to-sequence problems are two gaps referred to:

· the gap between the objective function and evaluation metric as the objective function is based on local, token-level prediction while the evaluation is mostly based on the general similarity between gold summaries and the generated ones

· the gap between training and predicting as a result of the accumulated error while generating summaries auto-regressively which is known as exposure bias

For pre-training developments, AdaptSum explored the effectiveness of pre-training influenced by pre-training domains for source domain, domain adaptive, and task adaptive based on BART model, then used RecAdam for optimization to tackle the large forgetting problem of continuing per-training [21]. And STEP model uses the idea of using input texts constructed from documents, pre-training focuses on reinstating the original document by three pre-training methods: reordering sentences, generating the next sentences of some sentences, and generating masked documents [22]. In another way, modifying inputs can also improve the abstractive summary quality. Moreover, using external signals as input to guide models is also a way of improvement.

For structure developments, Khandelwal et al. proposed a model using a decoder-only network that has a single pre-trained transformer to avoid copies, reduce the number of parameters, and ensure all parameters including attention weights are pre-trained [23]. Another model named PRIMERA, which is designed for multi-document summary uses a Longformer Encoder-Decoder framework with a model struct for multi-document input and pre-training on entity pyramid using gap-sentence-generation [24]. The two-staged transformer-based model separates the source into segments and uses each segment to generate extractive summaries in the first stage and use extracted summaries to generate abstractive one in second stage [25].

Developments can also be done by combining existing models with models from other machine learning areas like meta-learning and reinforcement learning. By using Model-Agnostic Meta-learning with two knowledge-rich corpora: large pre-training models and various existing corpora, the past experiences are leveraged to tackle limited labeled examples [26]. Apart from that, another model, RefSum also uses meta-learning but aims at selecting the best summary from many candidates generated

DOI: 10.54254/2755-2721/5/20230543

from some baseline models [27]. Besides, one possible way to solve the problems mentioned above is to employ the Actor-critic method in reinforcement learning with the sequence-to-sequence attention frameworks as an actor and maximum likelihood estimation plus global summary quality as a critic [28]. Another model using reinforcement learning is GOLD using off-policy reinforcement learning for text generation [29]. Apart from these, hybrid many models can also have great development, HATS hybrids more than two models to imitate different stages of human-like reading strategy: a knowledge-based attention network to simulate knowledge before reading, a multi-task encoder-decoder network that includes a pre-training model to imitate reading, and a generative adversarial network to imitate thinking procedure, and does not have the problem of large space of sequence but sparse rewards for close to references of reinforcement learning [30]. And combining sequence-to-sequence model and saliency model is also a possible form to assist the decoder to find important tokens [31].

Apart from those, optimizing the objective function is another direction for improvement, especially for reducing the gap from the objective function to the evaluation metric. Another model uses inconsistency loss to punish the inconsistency between word level and sentence level attention [32]. Adding contrastive learning parts in the loss function, SeqCo treats the original text, gold summary, and the generated summary as various aspects of the same text to maximize their similarity of them [33], and ConSum alleviates the exposure bias that was discussed in previous sections [34].

The most modern method is to generate many candidate summaries and then use text ranking to select the best one. For ranking strategies, SimCLS uses the sequence-to-sequence model in the procedure generating candidate summaries, RoBERTa as a pre-training model, and contrastive learning for ranking which can also avoid the inconsistency of training and evaluation [35]. In contrast, BRIO model uses BART and adjusts beam search for candidate generation and contrastive loss for selecting the best summary [36]. Moreover, SummaReranker is just a multi-task ranker for the abstractive summary which can use different evaluation methods for a classifier with a mixture-of-experts structure [37].

Table 2. ROUGE – 1, 2, L scores for models in section3 using CNN/DM dataset.

	R-1	R-2	R-L
AdaptSum [21]	35.07	-	-
STEP [22]	43.75	20.81	41
summarization using a single pre-trained transformer [23]	39.65	17.74	38.65
PRIMERA [24]	-	-	-
two-staged transformer-based model [25]	-	-	-
MTL-ABS [26]	-	-	-
RefSum [27]	44.96	21.50	41.43
AC-ABS [28]	44.96	21.50	41.43
GOLD-s [29]	44.82	22.09	41.81
HATS [30]	42.16	19.17	38.35
CIT+SE [31]	42.80	22.53	42.48
end2endw/inconsistency loss [32]	40.68	17.97	37.13
SeqCo [33]	45.02	21.80	41.75
ConSum [34]	44.53	21.54	41.57
SimCLS [35]	46.67	22.15	43.54
BRIO [36]	47.78	23.55	44.57
SummaReranker (PEGASUS) [37]	47.04	22.32	43.72

DOI: 10.54254/2755-2721/5/20230543

2.4. Dataset

XSum BBC articles and accompanying single sentence summaries which close to original text **CNN/D**, a news summarization dataset with 92579 data, which has 286817 training, 13368 validation, and 11487 test samples, each summary has 3.72 sentences on average.

Newsroom, includes 1.3 million human-written summaries of newsrooms articles of the major of 38 news publications from 1998 to 2017

New York Times annotated corpus, about 650k abstractive summaries for articles from the New York Times magazine from 1987, to 2007.

XL-Sum, multilingual abstractive summarization which includes about 44 different languages, extracted from BBC

Gigaword, produced by LDC (Linguistic Data Consortium), consists of around 4 million articles, for default it has 3803957 training samples, 189651 validation samples, and 1951 test samples.

DUC (Document Understanding Conference), available for version from 2001 to 2007, different version has different timeline and sizes.

Table 3. Datasets and splits if found.

dataset	#training	#evaluation	#test	resource	types	Published (newest)
xSum	203k	11k	11k	BBC articles	more extracted	2018
CNN/DM	287k	13k	11k	CNN, Daily Mail articles	more extracted	2016
Newsroom	995k	108k	108k	social media	abstracted	2021
XL-Sum	306k	11k	11k	BBC articles	abstracted	2018
Gigaword	380k	189k	1k	NYT, APW	abstracted	2003
New York Times annotated corpus	-	-	-	New York Times	more extracted	
DUC	203k	11k	11k	-	abstracted	2007

2.5. Evaluations

For evaluation, different metrics have various use cases, the most common one, ROUGE for Recall Oriented Understudy for Gisting Evaluation compares the gold summary and the generated one with ROUGE-N like the formula below. Apart from that, ROUGE-L for longest common subsequence is also commonly used in evaluations.

$$ROUGE - N = \frac{\sum_{S \in \{ReferencesSummaries\}} \sum_{grams_n \in S} Count_{match}(grams_n)}{\sum_{S \in \{ReferencesSummaries\}} \sum_{grams_n \in S} Count(grams_n)} \quad (2)$$

BLEU is based on precision and recall. And METEOR focuses on modified precision and recall [38].

To sum up, for NLG tasks, sequence-to-sequence with attention is used by many models and has many well-performed variants based on BERT and other pre-training models which is the focus of this article. There are many ways to improve the summary quality, including exploring pre-training, and structure, combining other methods, evaluation, and ranking, with ranking and combining other models, especially contrastive learning and reinforcement learning as the most modern approach. For datasets, the most common one is CNN/DM which is used for almost all the baseline models. And the most common evaluation function is ROUGE for 1, 2 grams, and ROUGE-L.

3. Conclusion

In this survey, a comprehensive overview of BERT-based pre-training models that can be used in an abstractive summary is conducted, including an introduction of BERT which is a typical pre-training model which starts a new era of NLP, baseline models based on BERT, problems, and developments of

DOI: 10.54254/2755-2721/5/20230543

them, datasets of these models, and evaluation methods. Several possible research directions are also suggested in the area of abstractive summary.

References

[1] Jones K S, 1992, "Natural language processing: an overview", in W. Bright (ed.) International encyclopedia of linguistics, New York: Oxford University Press, Vol. 3, 53–59.
[2] Elman Jeffrey L, Finding Structure in Time. Cognitive Science. 1990, 14 (2): 179–211.
[3] Hochreiter S and Schmidhuber J, 1997. Long short-term memory. Neural computation, 9(8), pp.1735-1780.
[4] Chung J, et al., 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*.
[5] Mikolov T, et al., 2013. Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781.
[6] Bahdanau D, Cho K. and Bengio, Y, 2014. Neural machine translation by jointly learning to align and translate.
[7] Radford A, et al., 2018. Improving language understanding by generative pre-training.
[8] Vaswani A, et al., 2017. Attention is all you need. Advances in neural information processing systems, 30.
[9] Devlin J, et al., 2018. Bert: Pre-training of deep bidirectional transformers for language understanding.
[10] Song K, et al., 2019. Mass: Masked sequence to sequence pre-training for language generation.
[11] Dong L, et al., 2019. Unified language model pre-training for natural language understanding and generation. Advances in Neural Information Processing Systems, 32.
[12] Bao H, et al., 2020, November. Unilmv2: Pseudo-masked language models for unified language model pre-training. In International Conference on Machine Learning (pp. 642-652). PMLR.
[13] Lewis M, et al., 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension.
[14] Qi W, et al., 2020. Prophetnet: Predicting future n-gram for sequence-to-sequence pre-training. arXiv preprint arXiv:2001.04063.
[15] Saleh M, et al., 2020, November. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. In International Conference on Machine Learning (pp. 11328-11339). PMLR.
[16] Wan D and Bansal M, 2022. FactPEGASUS: Factuality-Aware Pre-training and Fine-tuning for Abstractive Summarization.
[17] Xu J, Zhang H, and Wang J, 2019. Pretraining-based natural language generation for text summarization.
[18] Raffel C, et al., 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. J. Mach. Learn. Res., 21(140), pp.1-67.
[19] Liu Y, 2019. Fine-tune BERT for extractive summarization. arXiv preprint arXiv:1903.10318.
[20] Yu T, Liu Z and Fung P, 2021. AdaptSum: Towards low-resource domain adaptation for abstractive summarization.
[21] Liu Y and Lapata, M, 2019. Text summarization with pre-trained encoders. arXiv preprint arXiv:1908.08345.
[22] Zou Y, et al., 2020. Pre-training for abstractive document summarization by reinstating source text.
[23] Khandelwal U, et al., 2019. Sample efficient text summarization using a single pre-trained transformer.
[24] Cohan A, et al., W, 2021. Primer: Pyramid-based masked sentence pre-training for multi-document summarization.

DOI: 10.54254/2755-2721/5/20230543

[25] Cheng H, et al., 2020. A two-stage transformer-based approach for variable-length abstractive summarization. IEEE/ACM Transactions on Audio, Speech, and Language Processing, 28, pp.2061-2072.
[26] Chen Y S and Shuai H H, 2021, May. Meta-Transfer Learning for Low-Resource Abstractive Summarization. In Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 35, No. 14, pp. 12692-12700).
[27] Liu Y, Dou Z Y and Liu P, 2021. Refsum: Refactoring neural summarization.
[28] Li P, Lam W and Bing L, 2018. Actor-critic-based training framework for abstractive summarization.
[29] Pang R Y and He H, 2020. Text generation by learning from demonstrations.
[30] Qu Q, et al., 2019, July. Exploring human-like reading strategy for abstractive text summarization. AAAI Conference, Artificial Intelligence (Vol. 33, No. 01, pp. 7362-7369).
[31] Nishida K, et al., J, 2020. Abstractive summarization with combination of pre-trained sequence-to-sequence and saliency models.
[32] Hsu W T, et al., 2018. A unified model for extractive and abstractive summarization using inconsistency loss.
[33] Xu S, et al., 2022, June. Sequence level contrastive learning for text summarization. AAAI Conference, Artificial Intelligence (Vol. 36, No. 10, pp. 11556-11565).
[34] Li W, and Sun S, 2021. Alleviating exposure bias via contrastive learning for abstractive text summarization.
[35] Liu P and Liu Y, 2021. SimCLS: A simple framework for contrastive learning of abstractive summarization.
[36] Radev D, et al., 2022. BRIO: Bringing Order to Abstractive Summarization.
[37] Ravaut M, Joty S and Chen N F, 2022. SummaReranker: A Multi-Task Mixture-of-Experts Re-ranking Framework for Abstractive Summarization.
[38] Andhale, N and Bewoor, LA, 2016, August. An overview of text summarization techniques. In 2016 international conference on computing communication control and automation (ICCUBEA) (pp. 1-7). IEEE.