IAENG International Journal of Computer Science

A Novel Model of Generative Automatic Text Summarization Based on BART

Yahui Wang, Qingxia Chang, Xuelei Meng

Abstract—**To obtain useful information accurately and quickly from the massive text information is the most urgent need for people nowadays. The text automatic summarization technology summarizes and condenses the given source text information and generates short texts that are concise, fluent and retain key information. This paper proposes a novel automatic summarization model - Automatic Summarization Model based on BART and Actor - Critic Algorithm (BART-ACA-AST) to realize the efficient text summarization processing. The ROUGE metric system is used to evaluate the similarity and correlation between the mechanically generated text summaries and the reference summaries to assess the quality of the listed models. BERTScore is used to appraise the semantic resemblance between the rewritten summary and the reference summary more concisely. The computing results demonstrate the excellent performance of the model. The method proposed in this article can serve as a reference for the Automatic text summarization work.**

Index Terms—**automatic text summarization, BART, actor-critic algorithm, ROUGE, BERTScore**

I. Introduction

THE volume of information on the Internet is growing at an exponential speed. How to obtain useful information accurately and quickly from the massive text information is the most urgent need for people nowadays. The text automatic summarization technology summarizes and condenses the given source text information and generates short texts that are concise, fluent and retain key information. This can not only alleviate the redundancy problem caused by search engine retrieval, but also effectively solve the defect of information overload. In recent times, following the rapid expansion of artificial intelligence, big data processing technologies and the advancement of natural language processing research, the issue of automatic text summarization has received widespread attention. The study of Automatic text summarization constitutes a significant research subject within the domain of natural language processing and is of great value for in-depth exploration.

Manuscript received August 6, 2024; revised December 31,2024.

This research was funded by the Soft Science Special Project of Gansu Province - Self - Chosen Project (No. 24JRZA104).

Y. Wang is a lecturer of the Foreign Languages School, Lanzhou Jiaotong University, Lanzhou, Gansu, 730070, China (corresponding author, phone: 86-13919010651; fax:86-0931-4938025; e-mail: wangyh_lzjtu@163.com).

Q. Chang is a lecturer of the Foreign Languages School, Lanzhou Jiaotong University, Lanzhou, Gansu, 730070, China (e-mail: 407503484@qq.com).

X. Meng is a professor of the Traffic and Transportation School, Lanzhou Jiaotong University, Lanzhou, Gansu, 730070, China (e-mail: mengxl_lzjtu@163.com).

II. Related Work

A large number of literatures have been conducted on automatic text summarization. Reference [1] gave a wide-ranging overview of Automatic Text Summarization (ATS) which is a downstream task of Natural Language Processing (NLP). Reference [2] utilized two transformer-based language models, BART and T5, on the CNN_dailymail dataset and determined that BART achieved a higher ROUGE-1 Score than T5. Reference [3] presented a detailed overview and different approaches on machine learning-based automatic text summarization model application perspectives. Reference [4] proposed a novel processing approach for ATS tasks, constructed on a clustering scheme. The scheme was supported by a genetic optimization algorithm. Reference [5] presented a novel BERT-LSTM-BiGRU (BLG) to get the meaning of the sentences. It can embed the words, encode and decode the hidden states. Reference [6] proposed an ATS method based on an optimized graph - based algorithm, which was called TextRank algorithm. It also hired the clustering method of K-Means. Reference [7] presented a novel model to realize the summarization work which took the attention mechanism as the basic elements. Reference [8] focused the evaluation of the importance of the sentence position for ATS. Reference [9] applied the ant colony algorithm in language recognition. Reference [10] proposed a framework that combined content tokens and mathematical knowledge concepts in whole procedures, which was called Knowledge-Graph-based Mathematical Topic Prediction and embedded entities from mathematics knowledge graphs, integrated entities into tokens in a masked language model.

At present, for natural language processing, common generative models include PGN (Pointer Generator Network), MASS (Masked Sequence to Sequence Pre-training), T5 and BART.

Reference [11] proposed an new approach to attain end-to-end context ASR with graph neural network (GNN) encodings. Reference [12] presented a generative adversarial network to solve the detection problem of instrument reading. The method assured the precision and the quality of the reading of the instrument. Reference [13] tried to integrated the visual data and the original network to generate the caption of the news pictures, using a new generation network with pointer. Reference [14] designed a new pointer network to learn the plan of content. The network is based on the classic Transformer. Reference [15] constructed an ATS model to lead the transformation process, which combined the knowledge graph theory.

Reference [16] brought out a new scheme for training the protein sequence, which included masked language modeling and a supplementary training task, which is the common family forecast. Reference [17] combined the LSTM with a Unet to construct a segmentation network. The network can serve better in the field of renal segmentation. Reference [18] used the masked and permuted pre-training model to realize sentiment analysis.

Reference [19] carried out a grapheme-to-phoneme conversion with T5 based model and achieved very high conversion accuracy. Reference [20] introduced the T5 into grammar selecting problem to generate high quality text and built a high performance learning model. Reference [21] proposed an ensemble-type neural question generation model based on T5, which greatly increased the number of generated questions. Reference [22] constructed a novel model based on T5 model to generate survey files, which included mainly the multiple selections.

Reference [23] focused the problem of pseudocode generation and designed a BART-based model. Reference [24] paid attention to the robustness of the sentiment prediction. An algorithm was designed based on the Transformer to solve the prediction challenges in ATS. Reference [25] utilized the BART to build a novel model to improve the precision in ATS.

This paper proposes a reinforced automatic summarization model - Automatic Summarization Model based on BART and Actor-Critic Algorithm (BART-ACA-AST) to realize the efficient text summarization processing. Under the framework of reinforcement learning, multiple iterations of the ACA are used to improve the extraction process. In this paper, the ROUGE metric system is used to evaluate the similarity and correlation between the automatically generated text summaries and the reference summaries to assess the quality of the listed models. BERTScore (evaluating text generation with Bidirectional Encoder Representations from Transformers) is used to evaluate the semantic similarity between the rewritten summary and the reference summary more concisely.

TABLE I
STATISTICS INFORMATION OF SUMMARIES

Summary[a]	NTS	ASTL	ANSST	ANSS	ASL
Reference summary	1200	583.22	25.68	3.56	49.46
Extracted summary	1200	583.22	25.68	4.97	140.37

[a]NTS is the number of test samples. ASTL is the average source text length. ANSST is the average number of sentences in the source text. ASL is the average summary length.

III. Problem Statement

In automatic summarization of long texts, the existing extractive methods extract original text sentences to form summaries. The length of the summaries is often several times that of the reference summaries. For example, the extractive model Refresh based on encoder-decoder is used to extract the test set samples in the dataset English Corpora (https://www.english-corpora.org/iweb/). The results are shown in Table I. The reference summaries are often shorter than the extracted summaries. On average, each reference summary sentence has 13.9 words, and each extracted summary sentence has 28.2 words on average. The average length of the extracted summary sentences is more than twice that of the average reference summary sentences, indicating that the method of directly extracting entire sentences as summaries will lead to verbose and not concise summary information.

Example 1 is a sample of the summary extracted by this model. The key information contained in the extracted summary is underlined. The extracted summary contains 10 short sentences, while the reference summary contains only 2 short sentences. The length of the extracted summary is 3 to 4 times that of the reference summary. Moreover, the same person's name repeats in the extracted summary, and the sentences are not coherent enough. Therefore, the summary should not only come from extraction. To obtain a refined, concise and fluent summary, further rewriting is required.

A- Example 1

The source text is:

A visitation will be held today for the young victim of a fatal crash in the city's west end. 19-year-old Danielle Schmoll (photo courtesy of Facebook) had just dropped her brother off at the London Hunt and Country Club around 5:15 a.m. Wednesday, and was about to head home to Ilderton when her vehicle collided with a white Dodge pick-up truck on Oxford near Sanatorium. Schmoll was extricated from her vehicle and rushed to hospital. Though initially listed in critical condition, she succumbed to her injuries a short time later.

The reference abstract is:

A visitation will be held, Schmoll vehicle collided.

The extracted summary is

A visitation will be held, crash in the city's west end, Danielle Schmoll dropped her brother off, Danielle Schmoll head home, vehicle collided with truck, Schmoll was extricated from vehicle, rushed to hospital, she succumbed to her injuries later.

Generative summaries allow new words or phrases in the summary, which are highly flexible and more coherent. However, when facing long text summaries, inputting the entire document affects the generation process, resulting in slow decoding. Usually, problems such as key information loss and inaccurate summaries occur. The experimental results show that when the input is the entire long document, the output summary cannot cover all the key information in the input long text. However, generative summaries have a good generation effect on short texts and can generate accurate and fluent summaries. Example 2 is a summary sample generated by the DSR model. The summary that generated by the model is similar with the reference one, and the summary is accurate, fluent and has strong readability. So we can not only get the important information from the key sentences of long texts, but also make the summary more fluent.

B- Example 2

The source text is:

Liberal candidate and incumbent Deb Matthews says her office received complaints from constituents who received a letter from the Tories mentioning a polling station that *wasn't in the area.*

The reference abstract is

Matthews says office received complaints from constituents, who received a letter from the Tories, a polling station that wasn't in the area.

The generative summary is

Matth received complaints from constituents who received a letter mentioning a polling station wasn't in the area.

In the process of summary generation, the key sentences are rewritten to obtain the summary. The key lies in how to extract the key sentences. The extraction results affect the quality of the generated summary. Extracting non-key sentences will lead to the summary being unable to summarize the core content of the article. For example, the attention learned in the generation task is constrained by the estimated saliency in the extraction task to enhance their consistency. After re-generation, the redundant information contained in the extractive summary is reduced, but there is no process of multiple iterations, resulting in the extraction of non-key sentences, so that the summary does not accurately grasp the core idea of the original text. Reinforcement learning can be adopted to solve this problem. Through multiple iterations of the A2C algorithm to improve the extraction process, the extractor can obtain more accurate sentences for rewriting.

IV. BART-ACA-AST

Under the framework of reinforcement learning, this paper proposes a novel model of generative automatic text summarization based on BART, as shown in Fig. 1.

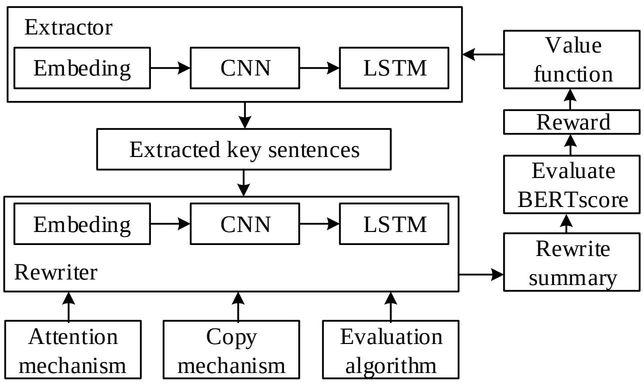

Fig. 1. Auto automatic summarization model based on reinforced learning

A new summary is obtained based on the processes of the extractor and the rewriter, serving as the actor of the A2C algorithm. The critic judges the score of the behavior of extracting key sentences by the actor based on the behavior, and the evaluator evaluates the semantic similarity between the rewritten summary and the reference summary through the BERTScore value as a reward. The actor then modifies the probability of selecting the behavior according to the score of the critic, ultimately improving the extraction process.

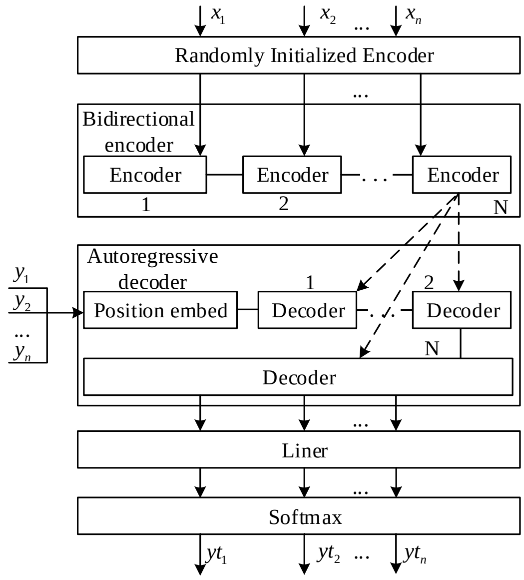

Fig. 2. Architecture of BART

A. Extractor

The extractor is based on the encoder-decoder framework, as shown in Fig. 2. The hierarchical document encoder reads the input sequence and generates sentence vectors combined with global contextual semantic information; the decoder combines the pointer mechanism to obtain the probability of each sentence in the original text being extracted, and at the same time extracts several key sentences to form a short text.

Since the extractor needs to extract the entire sentences of the source text, the model input requires sentence vectors to represent each sentence in the source text. The sentence vectors of each sentence in the source text are obtained by using the temporal convolutional model. Specifically, for each sentence in the source text, a sentence matrix is obtained from the word vectors of each word in the sentence. Through convolution, nonlinear activation and max pooling operations, the convolutional network outputs the sentence vector r_i of the *ith* sentence.

To combine the context information of a document and get the dependencies between sentences which is prolonged, the sentence vector r_i is sent to the bidirectional Long Short-Term Memory (LSTM) network for training to generate a sentence vector hi containing contextual semantic information, and then hi is used to represent the *ith* sentence in the source text.

After obtaining the sentence vector h_i, a pointer network based on LSTM is used as the decoder to select the key sentences. For each extraction time step t, the probability that each sentence in the source text is extracted is:

$$\varsigma_i^t = \begin{cases} \xi_p^T \cdot \tanh(W_{p1}h_i + W_{p2}e_t), i_t \neq i_k, \forall k < t \\ -\infty,\ else \end{cases} \tag{1}$$

$$P(i_t | i_t, i_t, ..., i_{t-1}) = soft\max(\varsigma^t) \tag{2}$$

where e_t is the output of the pointer network at the extraction time step t, specifically as follows.

$$\varepsilon_i^t = \varsigma_g^T \cdot \tanh(W_{g1}h_i + W_{g2}z_t) \tag{3}$$

$$\alpha_i^t = soft\max(\varepsilon^t) \tag{4}$$

$$e_t = \sum_i \varepsilon_i^t \cdot W_{g1} \cdot h_i \tag{5}$$

where z_t is the output of the single-layer LSTM, and both W and v are learnable parameters. The retrieval model can be seen as a bivariate classification problem, which means that the sentences can be divided into two groups- the summary sentences and the non-summary sentences.

B. Rewriter

To obtain a concise and fluent summary, after the extractor extracts the key sentences to form a short text, the rewriter performs generative rewriting on the short text. The rewriter adopts the standard encoder-decoder model based on the attention mechanism, with the input short text $x = \{x_1, x_2, ..., x_M\}$ and the output summary sequence $y = \{y_1, y_2, ..., y_N\}$. To consider the context state of the current word, the encoder adopts the Bidirectional Long Short-Term Memory network (Bidirectional Long Short-Term Memory network, Bi-LSTM). The attention mechanism is introduced to enable selectively reading the encoded semantic vector information during decoding. For the problem of Out-Of-Vocabulary (OOV) words, the copy mechanism [26] is adopted to solve it.

C. Reinforcement learning

In the process of long text summarization, multiple iterations of the extraction process are carried out, and the actor-critic algorithm is adopted. Compared with the REINFORCE algorithm [27], the actor-critic algorithm adds a critic network, which can solve the problem of high variance and thereby effectively improve the quality of the extracted key sentences.

Since the traditional summary evaluation metric ROUGE can only calculate the *n*-gram phrase overlap between the candidate summary and the reference summary and cannot capture the semantic relationship of the summary. Since key sentences need to be rewritten, an evaluator is adopted to calculate the BERTScore value of the rewritten summary and the reference summary as the semantic-based reward in reinforcement learning to improve the quality of the sentences extracted by the extractor.

BERTScore utilizes the pre-trained contextual embeddings from BERT (Bidirectional Encoder Representations from Transformers) and matches the words in the reference summary and the candidate summary through cosine similarity, and is mentioned as [28]:

$$v_R = \frac{\sum_{w_i \in w} f_{id}(w_i) \max_{\hat{w}_j \in \hat{w}} W_i^T \hat{W}_j}{\sum_{w_i \in w} f_{id}(w_i)} \tag{6}$$

$$v_P = \frac{\sum_{w_j \in w} f_{id}(w_j) \cdot \max_{\hat{w}_i \in \hat{w}} W_j^T \hat{W}_i}{\sum_{w_j \in w} f_{id}(w_j)} \tag{7}$$

$$v_F = \frac{2 \cdot V_R \cdot V_P}{V_R + V_P} \tag{8}$$

W and $\hat{W}$ represent the contextual word embeddings from BERT of the word m in the reference summary and the word m̂ in the candidate summary. The function $f_{id}(\cdot)$ is used to calculate the inverse document frequency.

The Actor-Critic Algorithm (ACA) is an important algorithm in reinforcement learning. It combines two parts: the actor and the critic. The "actor" is responsible for taking actions based on the strategy and attempts to optimize the strategy to obtain more rewards. The critic is responsible for evaluating the value of the actions taken by the actor. During the training process, the actor collects empirical data by interacting with the environment. The critic updates the value estimation based on these data, and then the actor adjusts the strategy according to the feedback from the critic. This process is continuously looped to gradually optimize the strategy and value estimation. ACA is adopted in this paper to improve the summarization quality. The actor refers to the policy function $\pi_\theta(c,a)$. At each extraction time step t, the current state of the extraction network is the set of the key sentences d_i that have been extracted. The actor observes the current state $s_t = (D, d_{v_{t-1}})$, and samples an action $v_t \sim \pi_\theta(s_t, v)$ according to the policy $\pi_\theta(c,a)$, and extracts the next sentence d_i from the original text D.

The critic network evaluates the BERTScore value of the currently rewritten summary sentence $g_{d_{v_t}}$ and the corresponding sentence in the reference summary b_t as the reward.

$$\Delta_{t+1} = BERTScore_{F_1}(g(d_{v_t}), b_t) \tag{9}$$

While one summary process is considered as one episode, the evaluation of the actual sampled behavior that satisfies the policy $\pi_\theta(c,a)$ can be obtained by backward calculation and discounting to get the total return R_t of the entire summary:

$$R_t = \sum_{t=1}^{N} \gamma^t \Delta_{t+1} \tag{10}$$

where N represents the number of extracted key sentences; γ is the discount factor which is a constant.

R_t is taken as the evaluation of function $Q^{\pi_\theta}(s,v)$. The ACA selects the state value function $V^{\pi_\theta}(s)$, which is only based on the state, as the baseline function, and obtains the Advantage function:

$$A^{\pi_\theta}(s,v) = Q^{\pi_\theta}(s,v) - V^{\pi_\theta}(s) \tag{11}$$

The Advantage function records the additional value of taking action v_t at state s_t compared to being at state s_t. This is consistent with the goal of policy improvement. The gradient of the policy objective function can be expressed as:

$$\Omega_\theta J(\theta) = E_{\pi_\theta}[\Omega_\theta \log \pi_\theta(s,v) \cdot A^{\pi_\theta}(s,v)] \tag{12}$$

The policy gradient obtained from the Advantage function can be positive or negative. When it is positive, extracting the sentence is encouraged. When it is negative, extracting the sentence is not encouraged. At this time, the loss function of the actor is:

$$L_\pi = -\log \pi_\theta(s,v) \cdot A^{\pi_\theta}(s,v) \tag{13}$$

The Mean Squared Error (MSE) is used as the loss function by the critic. θ is updated due to the following rule:

$$\theta \leftarrow \theta + \alpha \cdot \Omega_\theta \cdot \log \pi_\theta(s,v) \cdot A^{\pi_\theta}(s,v) \tag{14}$$

V. Computing Case and Results Analysis

A. Basic data of the computing case

The experimental data is from the iWeb sub-database of English Corpora dataset, as shown in Table II. This dataset contains 382,127 pairs of training sets, 17,860 pairs of validation sets, and 15,368 pairs of test sets. In this paper, the training set is used to train the model, multiple models with better training effects are saved, and the validation set is used to select the optimal model, and the final result is obtained by testing on the test set.

TABLE II
DATASET OF COMPUTING CASE

Dataset[a]	NTS	ASTL	ANSST	ANSS	ASL
English Corpora-iWeb	41535 (382127 /17860 /15368)	754	28.48	3.71	52

[a]NTS is the number of test samples. ASTL is the average source text length. ANSST is the average number of sentences in the source text. ASL is the average summary length.

B. Evaluation indicator

The evaluation of automatic summaries adopts the common summary evaluation criterion ROUGE [29]. The evaluation on the quality of the summary is to calculate the N-gram phrase equivalence between the reference summary and the prospective summary. The ROUGE evaluation metric mainly consists of ROUGE-n, where n is a positive integer, often be set as 1, 2, 3…, etc. ROUGE-1 (unigram) and ROUGE-2 (bigram) are adopted to measure the richness of summary information. ROUGE-L is also an important evaluation index, which represents the longest common subsequence of the reference summary and the candidate summary, usually used to measure the fluency of the summary content.

In this paper, ROUGE-1 (unigram), ROUGE-2 and ROUGE-L are used to measure the quality of the summaries.

C. Parameter setting

For all the experiments, word vectors are trained based on word2vec, the dimension of the word vector is 128, the dimension of the hidden unit vector of LSTM is 256, and there are 20,000 words in the word table. In the training stage, the Adam optimizer is used to train the model. The learning rate is set to 0.001, the momentum parameters β_1 is set to be 0.9, β_2 is set to be 0.999 and $\varepsilon = 10^{-8}$, the discount factor is set to 0.95, the batch-size is set to be 32, and Beam search with a size of 5 is used during decoding.

D. Comparison models

To verify the performance of the summaries generated by BART-ACA-AST model, the currently well-performing extractive models and generative models are selected for comparison and verification. The details of each comparison model are PGN, MASS and T5.

PGN can copy words from the source text while generating new words through the pointer mechanism, effectively alleviating the problem of inconsistency between the generated content and the original text. For example, it can accurately quote professional terms when dealing with scientific and technological articles.

MASS model for summary generation improves the quality and coherence of the generation. When dealing with news reports, it can generate concise and clear summaries.

T5 is a powerful automatic summarization model. The core advantage of the T5 model lies in its unified text-to-text framework, which enables it to handle multiple natural language processing tasks in a universal way, including automatic summarization. Compared with other models, T5 shows strong generalization ability when dealing with texts in different fields and styles. Whether it is text in the fields of technology, humanities, or entertainment, it can generate summaries of high quality.

E. Comparison experiment results

The traditional evaluation criterion ROUGE is adopted to evaluate the results of the comparison experiments, as shown in Table III, which is divided into three groups: extractive models, generative models, and the model in this paper. The highest score of the indicators is marked in bold numbers. It can be seen from Table III that the model in this paper has the best performance among all models in the ROUGE-1 and ROUGE-L scores, indicating that the overall performance of this model is the best. Compared with the extractive model PGN, the model in this paper has increased by 0.54, 0.99 and 0.34 percentage points respectively in the ROUGE-1, ROUGE-2 and ROUGE-L indicators, see Table III. Compared with the MASS model, when using the BART-ACA-AST model for calculation, the three parameters have increased by 0.58, 0.68, and 0.36 respectively. And, compared with T5 model, the model in this paper has increased by 0.3, 0.05, and 0.3 percentage points respectively in the ROUGE-1, ROUGE-2, and ROUGE-L indicators, indicating that the BART model based on reinforcement learning has better informativeness and fluency of the summary.

Analysis of the Similarity Improvement Ability

It can be seen that the BART-ACA-AST model comprehensively surpasses the PGN, MASS, and T5 models in terms of the ROUGE-1, ROUGE-2 and ROUGE-L.

The R-AVG score is the average of ROUGE-1, ROUGE-2, and ROUGE-L. The R-AVG score of the model in this paper

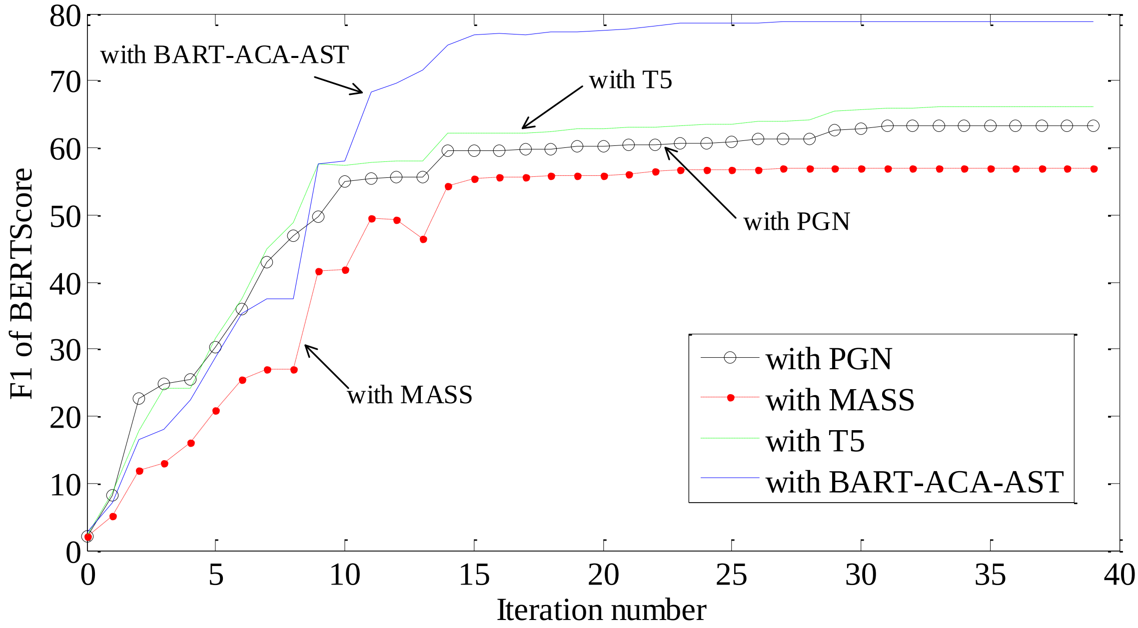

Fig. 3. Comparison of the overall performance of four models

is 31.67, achieving the best effect compared with the baseline model. However, the R-AVG values obtained with the PGN, MASS, and T5 models are 31.04, 31.12, and 31.45 respectively. The value of R-AVG obtained with BART-ACA-AST has increased by 2.03%, 1.77%, and 0.70% respectively, compared that with using PGN, MASS, and T5.

Based on the above comparison experiments, it can be known that the performance of the model in this paper has achieved a certain improvement on the similarity between the generated text and the reference text.

Analysis of the Precision and Recall

By applying the model proposed in this paper, the P value has increased by 16.68, 17.3, and 13.52 respectively compared with the results obtained by applying PGN, MASS, and T5 (see Table IV), indicating that the newly proposed model has a significant improvement in terms of precision.

And the R value is 16.28, 16.51, and 10.26 higher respectively than the results obtained by applying PGN, MASS, and T5, proves that the new model has high sensitivity and a high ability to correctly identify positive examples.

Compared with PGN, MASS, T5 models, the F1 value of the model in this paper has increased by 16.53, 16.94 and 11.83 percentage points, showing that the overall performance of the novel model is better than that of PGN, MASS, and T5.

The comprehensive results of the ROUGE index and the BERTScore index indicate that the presented method can improve the semantic information of the summary, and the informativeness and fluency of the summary have also been further improved.

TABLE III
COMPUTING RESULTS OF ROUGE INDEXES

Model	ROUGE-1	ROUGE-2	ROUGE-L	R-AVG
PGN	39.47	17.30	36.36	31.04
MASS	39.43	17.61	36.34	31.12
T5	39.71	18.24	36.40	31.45
BART-ACA-AST	40.01	18.29	36.70	31.67

TABLE IV
COMPUTING RESULTS OF BERTSCORE INDEXES

Model	P	R	F1
PGN	62.54	63.77	63.21
MASS	61.92	63.54	62.80
T5	65.70	69.79	67.91
BART-ACA-AST	79.22	80.05	79.74

Fig. 3 shows that the F1 value of BERTScore tends to be stable when the calculation number reaches 30. It is 63.21 with PGN, 62.80 with MASS, 67.91 with T5 and 79.74 with BART-ACA-AST. And we can see that the performance of BART-ACA-AST is superior to other models, not only in the final optimization result but also throughout the entire optimization calculation process.

Analysis of the Computational Efficiency

As can be seen from Table V, the BART-ACA-AST model consumes the least total time in the calculation process, which is 3405.6 seconds. It is reduced by 197.1 seconds, 191.6 seconds, and 121.8 seconds respectively compared with the PGN, MASS, and T5 models.

TABLE V
COMPUTING RESULTS OF COMPUTING TIME

Model	Total time	Iteration number	Average time
PGN	3602.7s	30	120.1s
MASS	3597.2s	28	128.5s
T5	3527.4s	29	121.6s
BART-ACA-AST	3405.6s	25	136.2s

And it can be seen that the BART-ACA-AST model consumes a relatively long average time in each iteration process. This is because in each iteration process, more complex optimization processing is carried out. However, precisely because of this, each calculation has better accuracy and recall rate, thus ensuring the overall performance of the model. The number of calculations to reach the optimal is 25 times, while the number of calculations of the other several models is 30, 28, and 29 respectively. Therefore, generally speaking, the model given in this paper is also superior in terms of computational efficiency to the current main models.

VI. Conclusion

In this paper, a novel model of generative automatic text summarization model based on BART is proposed. This model extracts key information from the original text based on the extractor of the CNN and RNN hybrid neural network, guides the rewriting of the rewriter based on the copy mechanism and the attention mechanism, uses reinforcement learning to connect the two networks, and combines the semantic-based reward method to train the entire model. Experimental results show that in the long text automatic summarization task, the A2C-RLAS model generates more accurate summary content, contains more key information of the original text, the summary language is more fluent, and effectively avoids the repetition of the generated content.

In addition, the model in this paper can be considered for further improvement: First, the pre-trained model BERT can be used for word embedding; then, to investigate the "factuality" of the generative summary, factual knowledge can be integrated into the model. Finally, the applicability of the model in this paper to new datasets in different scenarios, such as the summaries of academic papers, can be investigated.

References

[1] Iskander Akhmetov, Sabina Nurlybayeva, Irina Ualiyeva, Alexandr Pak, and Alexander Gelbukh, "A Comprehensive Review on Automatic Text Summarization," Computacion y Sistemas, vol. 27, no.4, pp1203-1240, 2016

[2] Ritika Rao, Sourabh Sharma, and Nitin Malik, "Automatic Text Summarization Using Transformer-based Language Models," International Journal of System Assurance Engineering and Management, vol. 15 , no.6, pp2599-2605, 2024

[3] P. Radhakrishnan, and G. Senthil kumar, "Machine Learning-based Automatic Text Summarization Techniques," SN Computer Science, vol. 4, no.6, 2023, Doi: 10.1007/s42979-023-02293-z

[4] Ángel Hernández-Castañeda, René Arnulfo Garcá-Hernández, Yulia Ledeneva, and Christian Eduardo Millán-Hernández, "Language-independent Extractive Automatic Text Summarization Based on Automatic Keyword Extraction," Computer Speech and Language, vol. 71, no.C, pp101267, 2022

[5] Sonali Mhatre, and Lata L. Ragha. "A Hybrid Approach for Automatic Text Summarization by Handling Out-of-vocabulary Words Using TextR-BLG Pointer Algorithm," Scientific and Technical Information Processing, vol. 51, no.1, 72-83, 2024

[6] Wenjun Liu, Yuyan Sun, Bao Yu, Hailan Wang, Qingcheng Peng, Mengshu Hou, Huan Guo, and Hai Wang, "Automatic Text Summarization Method Based on Improved TextRank Algorithm and K-Means Clustering," Knowledge-Based Systems, vol. 287, no.C, pp111447, 2024

[7] Abd Algani, and Yousef Methkal, "A Novel Deep Learning Attention Based Sequence to Sequence Model for Automatic Abstractive Text Summarization," International Journal of Information Technology (Singapore), vol. 16, no.6, pp3597-3603, 2024

[8] Griselda Areli Matias Mendoza, Yulia Ledeneva, and René Arnulfo Garcá-Hernández, "Determining the Importance of Sentence Position for Automatic Text Summarization," Journal of Intelligent & Fuzzy Systems, vol. 39, no.2, pp2421-2431, 2020

[9] Jinghui Li, "Research on Image Segmentation Technology and Intelligent Language Recognition Based on Ant Colony Algorithm," IAENG International Journal of Computer Science, vol. 50, no.2, pp553-563, 2023

[10] Canghong Jin, Wenkang Hu, Hao Wu, Yabo Chen, and Minghui Wu, "Enhanced Language Model with Hybrid Knowledge Graph for Knowledge Topics Identification in Questions," IAENG International Journal of Computer Science, vol.49, no.4, pp1169-1177, 2022

[11] Guangzhi Sun, Chao Zhang, and Philip C. Woodland, "Graph Neural Networks for Contextual ASR With the Tree-constrained Pointer Generator," IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 32, pp2407-2417, 2024, Doi:10.1109/TASLP.2024.3389645

[12] Ye Lin, Zhezhuang Xu, Meng Yuan, Dan Chen, Jinyang Zhu, and Yazhou Yuan, "Pointer Generation and Main Scale Detection for Occluded Meter Reading Based on Generative Adversarial Network," Measurement: Journal of the International Measurement Confederation, vol. 234, no.0, pp114836, 2024

[13] Jingqiang Chen, and Hai Zhuge, "A News Image Captioning Approach Based on Multimodal Pointer-generator Network," Concurrency and Computation: Practice and Experience, vol. 34, no.7, pp5721, 2022

[14] Bayu Distiawan Trisedya, Jianzhong Qi, Haitao Zheng, Flora D. Salim, and Rui Zhang, "TransCP: A Transformer Pointer Network for Generic Entity Description Generation with Explicit Content-planning," IEEE Transactions on Knowledge and Data Engineering, vol. 35, no.12, pp1-12, 2023

[15] Weidong Li, Rong Peng, Yaqian Wang, and Zhihuan Yan, "Knowledge Graph Based Natural Language Generation with Adapted Pointer-generator Networks," Neurocomputing, vol. 382, no.C, pp174-187, 2020

[16] Seonwoo Min, Seunghyun Park, Siwon Kim, Hyun-Soo Choi, Byunghan Lee, and Sungroh Yoon, "Pre-training of Deep Bidirectional Protein Sequence Representations with Structural Information," IEEE Access, vol. 9, pp1, 2021, Doi: 10.1109/ACCESS.2021.3110269

[17] Gaoyu Cao, Zhanquan Sun, Chaoli Wang, Hongquan Geng, Hongliang Fu, Lin Sun, and Jiao Nan, "S2S-ARSNet: Sequence-to-Sequence Automatic Renal Segmentation Network," Biomedical Signal Processing and Control, vol. 79, no.Part 1, pp104121, 2023

[18] Nicole Kai Ning Loh, Chin Poo Lee, Thian Song Ong, and Kian Ming Lim, "MPNet-GRUs: Sentiment Analysis with Masked and Permuted Pre-training for Language Understanding and Gated Recurrent units, " IEEE Access, vol. 12, pp74069-74080, 2024, Doi: 10.1109/ACCESS.2021.3110269

[19] Markéta Řezáčková, Daniel Tihelka, and Jindřich Matoušek, "T5G2P: Text-to-text Transfer Transformer Based Grapheme-to-phoneme Conversion," IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 32, pp3466-3476, 2024, Doi: 10.1109/TASLP.2024.3426332

[20] Peerawat Chomphooyod, Atiwong Suchato, Nuengwong Tuaycharoen, and Proadpran Punyabukkana, "English Grammar Multiple-choice Question Generation Using Text-to-Text Transfer Transformer," Computers and Education: Artificial Intelligence, vol. 5, pp100158, 2023, Doi: 10.1016/j.caeai.2023.100158

[21] Myeong-Ha Hwang, Jikang Shin, Hojin Seo, Jeong-Seon Im, Cho Hee, and Chun-Kwon Lee, "Ensemble-NQG-T5: Ensemble Neural Question Generation Model Based on Text-to-text Transfer Transformer," Applied Sciences (Switzerland), vol. 13, no.2, pp903, 2023

[22] Ricardo Rodriguez-Torrealba, Eva Garcia-Lopez, and Antonio Garcia-Cabot, "End-to-End Generation of Multiple-Choice Questions Using Text-to-Text Transfer Transformer Models," Expert Systems with Applications, vol. 208, no.Suppl C, pp118258, 2022

[23] Anas Alokla, Walaa Gad, Waleed Nazih, Mustafa Aref, and Abdel-badeeh Salem, "Pseudocode Generation from Source Code Using The BART Model," Mathematics, vol. 10, no.3967, pp3967, 2022

[24] Ehtesham Hashmi, Sule Yildirim Yayilgan, and Sarang Shaikh, "Augmenting Sentiment Prediction Capabilities for Code-mixed Tweets with Multilingual Transformers," Social Network Analysis and Mining, vol. 14, no.1, 2024, Doi: 10.1007/s13278-024-01245-6

[25] Andrea Stevens Karnyoto, Chengjie Sun, Bingquan Liu, and Xiaolong Wang, "TB-BCG: Topic-Based BART Counterfeit Generator for Fake News Detection", Mathematics, vol.10, no.4, pp585, 2022

[26] Qiji Zhou, Donghong Ji, Yafeng Ren, and Hao Tang, "Dual-copying Mechanism and Dynamic Emotion Dictionary for Generating Emotional Responses," Neurocomputing, vol. 454, no.Suppl C, pp303-312, 2021

[27] RONALD J. WILLIAMS, "Simple Statistical Gradient-following Algorithms for Connectionist Reinforcement Learning," Machine Learning, vol. 8, no.3-4, pp229-256, 1992

[28] Shichao Sun, Ruifeng Yuan, Wenjie Li, Ziqiang Cao, and Sujian Li. "Dialogue Acts Enhanced Extract–abstract Framework for Meeting Summarization", Information Processing & Management, vol. 61, no.3, pp103635, 2024
[29] Jonathan Rojas-Simón, Yulia Ledeneva, and René Arnulfo García-Hernández, "Evaluation of Text Summaries without Human References Based on The Linear Optimization of Content Metrics Using A Genetic Algorithm", Expert Systems with Applications, vol. 167, no.C, pp113827, 2021

Y. Wang received the MA degree in Foreign Linguistics and Applied Linguistics in School of Foreign Languages, Lanzhou Jiaotong University in 2012. She is a lecturer in Lanzhou Jiaotong University. Her research interests are in the areas of scientific and technical translation, translation theory and practice.