Proceedings of Machine Learning Research vol 245:1–9, 2024 2024 International Conference on Machine Learning and Intelligent Computing

Attention-Enhanced Pointer Network for Summarization with Key Information

Qiming Li* 2021107272@ZUT.EDU.CN
Zhongyuan University of Technology, Zhengzhou, 451100, China

Liang Gao
Zhongyuan University of Technology, Zhengzhou, 451100, China

Editors: Nianyin Zeng and Ram Bilas Pachori

Abstract

Addressing the limitations of mainstream generative text summarization models, such as poor semantic quality, inappropriate allocation of weights to key information, and constraints in extracting the semantic essence of textual content by existing natural language generation models, we propose an Attention-Augmented Pointer Generation Network (AUPT). This model utilizes TextRank technology to extract crucial information, combines positional encoding with an adaptive masking mechanism to enhance positional attention scores, emphasizing the importance of key information in the text's semantics. Furthermore, by integrating the T5-Pegasus model with the pointer generation network, it effectively handles unknown vocabulary and replication issues, enabling more accurate and reliable semantic representations.

Keywords: Text summarization, T5-Pegasus pre-trained model, Key information, Pointer network, Transformer architecture, Attention mechanism

1. Introduction

In today's rapidly expanding information landscape, the demand for efficient extraction of essential information from vast textual datasets, including comments, news articles, and research papers, has become increasingly urgent. To address this demand, text summarization technology has emerged as a pivotal area of research within the field of natural language processing. Text summarization endeavors to leverage advanced machine learning and artificial intelligence algorithms to distill key insights from large volumes of textual data, thereby assisting users in saving time and cognitive effort.

Text summarization, a critical research domain in natural language processing, was first systematically explored by Luhn (1958) Currently, text summarization methods are primarily categorized as generative and extractive approaches.

In order to address the limitations of text extraction-based summarization and the challenges associated with generative abstract methods, Rush et al. (2015) introduced a fully data-driven approach to abstract sentence summarization. This approach leverages a locally attentive neural network language model, which generates corresponding content for each summary word based on input sentences. Notably, this strategy led to significant performance improvements.

To mitigate token repetition issues in Seq2Seq models (Sutskever et al., 2014) , Gu et al. (2016a) incorporated a copy mechanism (Gu et al., 2016b) into neural network-based Seq2Seq learning, introducing a novel encoder-decoder model called CopyNet. CopyNet effectively combines conventional vocabulary generation within the decoder with the new copy mechanism. The introduction

of the Transformer model marked a pivotal moment in pre-trained models. Vaswani and his team (Vaswani et al., 2023) proposed a novel network architecture entirely based on attention mechanisms, eliminating the need for recursion and convolutional neural networks (LeCun et al., 1989) . This model exhibited superior quality and enhanced parallelism, resulting in reduced training time. The fusion of sequence-to-sequence architecture and attention mechanisms significantly improved the quality of summaries. With the consideration of the copy mechanism, See et al. (2017) introduced a summary approach based on a pointer-generator network. This approach amalgamates the generative capabilities of neural networks with the extraction capabilities of pointer networks, enabling the generation of accurate and coherent summaries. Furthermore, this method introduces a coverage mechanism (Tu et al., 2016) to prevent the generation of repetitive summaries.

Jianlin (2021) released a Chinese generative pre-training model named T5 PEGASUS, based on the Google mT5 model. To address the issue of the original mT5 model's SentencePiece tokenizer being less user-friendly for Chinese, the tokenizer was switched back to BERT's tokenizer with the addition of segmentation functionality, further enhancing the vocabulary. For pre-training tasks, Su followed the approach of the PEGASUS model, training a Seq2Seq model by constructing data pairs similar to abstracts. This improves T5 PEGASUS's performance in Chinese text generation tasks, enhancing its overall quality and utility.

1.1. Contributions

In this work, we have made significant contributions in two key areas:

(1) Enhanced Information Integration: We have introduced a novel approach that combines the strengths of BiLSTM (Schuster and Paliwal, 1997) and TextRank methods. This fusion enables the model to adaptively learn the importance of different positions within the text and incorporate this knowledge into a multi-head attention mechanism. As a result, the multi-head attention mechanism becomes more effective in assigning scores to crucial textual information. This enhancement significantly improves the encoder's ability to capture semantic information efficiently.

(2) Pointer Mechanism Integration: We have integrated the Pointer mechanism with the T5-Pegasus pre-trained model. This integration empowers the model to handle challenges posed by unknown vocabulary and content repetition. By allowing the model to directly copy content from the input sequence, it becomes better suited to tackle complex output scenarios. This adaptation not only enhances the model's generalization capabilities and flexibility but also effectively reduces the size of the output vocabulary.

2. Materials and Methods

To address the limitations of current natural language generation models in extracting essential semantic information from text, we propose an advanced model based on the T5-Pegasus architecture. This model, named Attention-Augmented Pointer Generation into Networks (AUPT), consists of three primary modules:

(1) Key Information Extraction Module: Within AUPT, the key information extraction module plays a pivotal role. It extracts relative position information of key content from the source text. This is achieved by processing the input through the Embedding layer, BiLSTM layer, and TextRank layer. The module constructs an enhanced mask matrix specific to the input batch. This matrix actively participates in the computation of Multi-Head Attention during the semantic information extraction phase conducted by the T5-Pegasus module.

(2) Attention Mechanism Encoder with Key Information Integration: The T5-Pegasus module, the attention mechanism encoder in AUPT, focuses on incorporating key information into the encoding process. It primarily utilizes a multi-head attention mechanism to capture deep global information from the source text. The integration of key information enhances the encoder's capacity to recognize crucial semantic details.

(3) Pointer Network Module: The pointer network module is a crucial component of AUPT. It employs a pointer generator model to calculate the generation probability at each time step in the decoding process. This probability dictates whether a word is generated from the vocabulary or directly copied from the source text, providing adaptability and control in complex text summarization scenarios. The model's architecture is visually represented in Figure 1 for better comprehension.

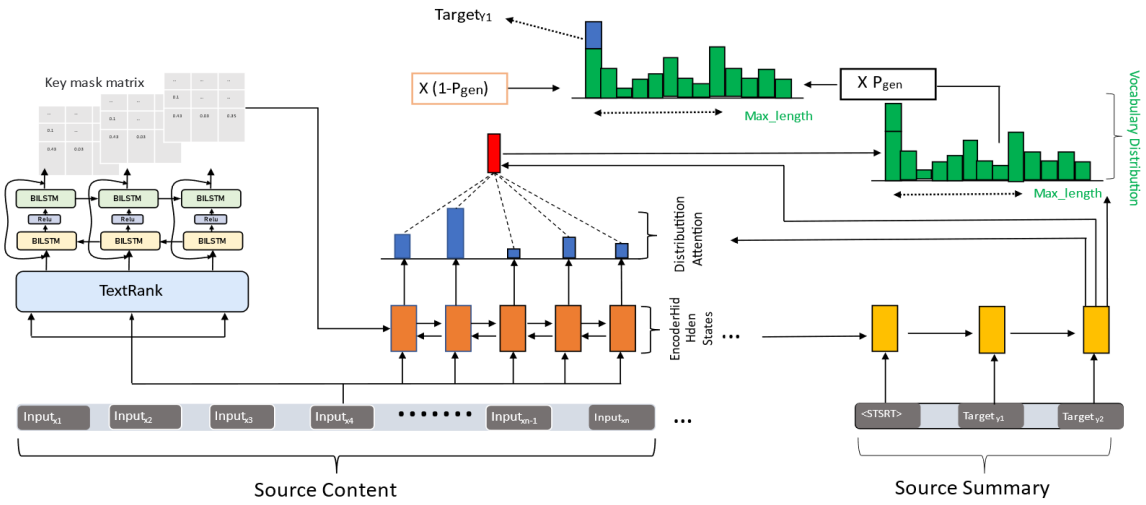

Figure 1: Overall framework.

2.1. Fusion of Key Information with a Transformer Encoder

To extract essential information from the content, we employ BiLSTM+TextRank. This process results in the creation of a mask matrix $mask$ denoted as $M_{ij} \in R(B \times N)$, where each M_{ij} element represents a specific entry in the matrix of key information. The construction of the key information matrix is detailed as follows:

$$m \in M \begin{cases} m{=}1 & important \\ m{=}0 & unimportant \end{cases} \tag{1}$$

After obtaining the matrix, key information captures the positional information corresponding to critical elements within the sequence. The positional markers associated with the key information are merged into the input while maintaining the same dimensionality. Additionally, we utilize positional information to fine-tune the weights of the attention mechanisms, effectively addressing issues arising from spatial heterogeneity and output variations. This process is illustrated in Figure 2.

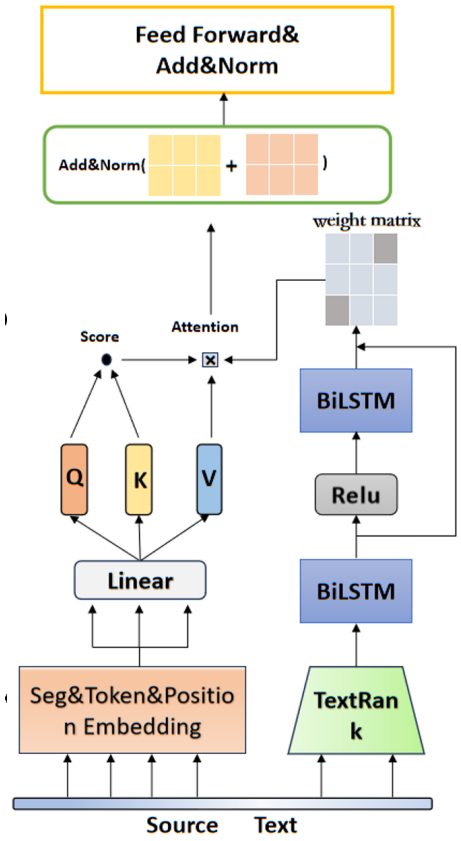

Figure 2: Flowchart of TextRank+BiLSTM algorithm.

The encoding stage of integrating the key information Transformer encoder mainly consists of the following components:

$$A_{raw} = \frac{Q_i K_i^T}{\sqrt{d/h}} \tag{2}$$

To begin, calculate the attention score matrix without any adjustments, denoted as A_{raw}. The query matrix Q_i is derived by passing the input through an embedding layer and subsequently a fully connected layer. Additionally, K_i^T is formed by taking the transpose of the last two dimensions of the key matrix, which is obtained through a similar process involving an embedding layer and a fully connected layer. A scaling factor is introduced, which is employed to reduce the overall magnitude of the data. This scaling factor serves the purpose of simplifying subsequent calculations within the softmax layer.

$$\overrightarrow{H} = (W_{hh}^{f(x)} Re(W_{xh}^{f(x)} M_x) + M_x) \tag{3}$$

Next, employ a residual connection to combine the enhanced attention matrix derived from the output of the fully connected layer with the unadjusted attention score matrix. The resulting output is denoted as $\overrightarrow{H}$, representing the hidden state following the residual connection. $W_{xh}^{f(x)}$ corresponds to the weight matrix of the first-layer fully connected neural network, while $W_{hh}^{f(x)}$ corresponds to the weight matrix of the second-layer fully connected neural network. The Re notation signifies the

application of the rectified linear unit $relu$ activation function operation, and M_x parameters refer to the position matrix of key information obtained from the BiLSTM and BiLSTM layers.

$$A_{\mathrm{masked}} = A_{res} + mask \tag{4}$$

$$Attention_{raw} = (A_{masked}\overrightarrow{H}) \tag{5}$$

Upon obtaining the hidden state output $\overrightarrow{H}$, we apply the matrix to the adjusted attention weight matrix, yielding the final computation of the multi-head attention mechanism. The resulting matrix, denoted as A_{masked}, represents the attention score matrix following the application of $mask$. This process involves the adjustment of attention weight matrices, the utilization of residual connections, and the integration of the multi-head attention mechanism, collectively enhancing the emphasis on crucial tokens. The adaptive adjustment of attention weights proves instrumental in capturing contextual information more effectively. Furthermore, the incorporation of residual connections serves to retain the original attention score information while amalgamating it with the enhanced attention data, thereby bolstering the model's representational capacity and refining the precision of attention distribution. Notably, AUPT exhibits adaptive adjustment capabilities during the summary generation process.

2.2. Integrating Pointer-Generator Networks with Transformer Decoders

Within the Transformer decoder, at each time step t, a combination of the Multi-Head Self-Attention and Multi-Head Encoder-Decoder Attention mechanisms is employed. Initially, we calculate self-attention weights and encoder-decoder attention weights to capture the relationships between the input sequence and the summary vocabulary generated so far. In each decoder layer, we amalgamate the outputs of self-attention and encoder-decoder attention with the input to the decoder. This is followed by the application of a Position-wise Feed-Forward Network to compute the output. This process iterates across multiple decoder layers to generate probabilities. Finally, in the output of the last decoder layer, we employ a linear layer followed by a $softmax$ activation function to compute the generation probabilities $p_{vocab}(y_t)$ for each word in the vocabulary.

$$p_{vocab}(y_t) = softmax(W_o Decoder(y_{t-1}, H) + b_o) \tag{6}$$

In the given context, W_o and b_o are used to represent the weights and biases of the linear layer. Meanwhile, $Decoder(y_{t-1}, H)$ signifies the decoder's output. Following this, the encoder-decoder attention weights are utilized to compute the probability $p_{vocab}(y_t)$ for each word from the input sequence to be replicated in the output sequence:

$$p_{copy}(y_t) = 1 - p_{vocab}(y_t) \tag{7}$$

Subsequently, the generation probability gating value $p_{gen} \in [0, 1]$ is calculated based on the decoder's hidden state and attention weights.

$$p_{gen} = \sigma(W_g[Decoder(y_{t-1}, H), a_t] + b_g) \tag{8}$$

In this context, σ represents the $sigmod$ activation function, while W_g and b_g correspond to trainable weights and biases, respectively. The final output probability distribution $p(y_t)$ is computed by merging the generation probability $p_{vocab}(y_t)$ with the pointer probability $p_{copy}(y_t)$.

$$p(y_t) = p_{gen} p_{vocab}(y_t) + (1 - p_{gen}) p_{\mathrm{copy}}(y_t) \tag{9}$$

Finally, a vocabulary word is sampled from the probability distribution $p(y_t)$ as the output for the current time step, which is then passed to the decoder as input for the next time step.

3. Experimental Study

3.1. Dataset

In this investigation, we employed two distinct datasets, namely NLPCC2017 (Qiu et al., 2017) and LCSTS (Hu et al., 2015), as the primary sources for our experimental evaluation. A total of 50,000 samples were utilized from each of the NLPCC2017 and LCSTS datasets to conduct our experiments.

3.2. Evaluation Metrics

In our experimental analysis, we utilized the ROUGE (Lin, 2004) metric to evaluate the quality of the generated summaries. ROUGE is a widely recognized and applied metric for assessing the performance of automatic text summarization systems. It quantifies the quality of generated summaries by quantifying the n-gram overlap between the generated summary and the reference summary. Specifically, we focused on three primary evaluation metrics: Rouge-1, Rouge-2, and Rouge-L.

3.3. Results and Analysis

In this section, we present the experimental results of our AUPT model on the dataset constructed for this study, employing the comparative models mentioned earlier. The results of these experiments are summarized in Table 1.

Table 1: Comparison of experimental result.

Model	LSTS			NLPCC		
	R-1	R-2	R-L	R-1	R-2	R-L
Lead-3(Nallapati et al., 2016)	28.32	15.24	25.63	28.24	15.32	25.51
RNN-context	29.95	17.47	27.22	29.88	17.52	27.32
SRB(Ma et al., 2017)	33.36	20.01	30.18	33.14	20.27	30.23
CopyNet	34.46	21.67	31.35	34.37	21.10	31.71
DGRB	37.03	24.22	34.24	37.37	24.25	34.63
Transformr	39.56	25.87	36.6	39.14	25.92	36.42
BERT[17]+Pointer	41.50	26.93	38.88	41.41	26.75	38.37
PEGASUS	42.62	27.07	41.15	42.76	27.29	41.68
AUPT	**43.58**	**28.43**	**41.92**	**43.49**	**28.40**	**41.47**

The experimental results demonstrate that Lead-3, which relies on fixed rules, has limited effectiveness in generating summaries for complex texts. RNN-Context enhances the ability to extract global semantics, but it faces limitations when processing long sequences. SRB improves the quality of summaries by leveraging semantic relevance and the Bi-GRU encoder. CopyNet introduces a copying mechanism, thereby improving the accuracy and readability of summaries. Transformer excels in processing long sequences and performing parallel computing due to its self-attention mechanisms. BERT-Pointer combines BERT's semantic representation with a pointer generation

mechanism, preserving both global semantic information and key details. Pegasus generates high-quality summaries by focusing on key information through a novel abstractive pre-training task.

3.4. Analysis of Component Sensitivity

To We conducted a series of ablation studies. Our baseline model was T5-Pegasus. The second model integrated a Pointer network with T5-Pegasus. Then, we combined the key information identified by the BiLSTM-TextRank method with T5-Pegasus, creating a third model, denoted as T5-BTR. Finally, we arrived at our proposed model, AUPT. The resulting experimental outcomes are presented in Table 2.

Table 2: Results of ablation experiments.

Model	LSTS			NLPCC		
	R-1	R-2	R-L	R-1	R-2	Ro-L
T5-Pegasus	42.62	27.50	41.02	42.60	27.47	40.08
T5-Pegasus+Pointer	43.06	21.23	40.50	43.12	17.5	27.3
T5-BTR	43.11	27.38	41.13	33.1	20.2	30.2
AUPT	**43.58**	**28.43**	**41.92**	**43.49**	**28.40**	**41.47**

During the ablation experiments, model performance gradually improved with the addition of components. Compared with the baseline model T5-Pegasus, the dual model incorporating the Pointer network significantly improved across evaluation metrics, indicating that the Pointer network effectively assisted the model in capturing critical information. The generative summarization approach, which combines T5-Pegasus and the Pointer network, proved more efficient than traditional methods. Furthermore, the triple model, which integrated key information extracted by BiLSTM-TextRank, demonstrated superior performance, confirming the crucial role of key information in the decoding process. The final model, which fused T5-Pegasus, the Pointer network, and key information extracted by BiLSTM-CRF, achieved optimal performance across all evaluation metrics, validating the effectiveness and rationality of the model structure designed for natural language processing tasks.

4. Conclusion

In this study introduces an attention-augmented pointer network-based summarization model, denoted as AUPT (Attention-Augmented Pointer Generation into Networks). Our approach leverages BiLSTM+TextRank techniques to perform deep key information extraction from the input text, facilitating the precise identification of critical semantic components. To determine the position of each key word within the sentence, we apply a positional encoding strategy and introduce an adaptive masking mechanism. Furthermore, our model seamlessly integrates the T5-Pegasus model and incorporates a Pointer mechanism during the decoding process. In rigorous Rouge evaluation metrics, the generated summaries consistently outperform the benchmark models, thus affirming the efficacy and soundness of our proposed model.

References

Jiatao Gu, Zhengdong Lu, Hang Li, and Victor O.K. Li. Incorporating copying mechanism in sequence-to-sequence learning. In Katrin Erk and Noah A. Smith, editors, *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1631–1640, Berlin, Germany, August 2016a. Association for Computational Linguistics. doi: 10.18653/v1/P16-1154.

Jiatao Gu, Zhengdong Lu, Hang Li, and Victor O.K. Li. Incorporating copying mechanism in sequence-to-sequence learning. In Katrin Erk and Noah A. Smith, editors, *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1631–1640, Berlin, Germany, August 2016b. Association for Computational Linguistics. doi: 10.18653/v1/P16-1154.

Baotian Hu, Zhengdong Lu, Hang Li, and Qingcai Chen. Convolutional neural network architectures for matching natural language sentences, 2015.

Su Jianlin. T5 pegasus: Open-sourcing a chinese generative pretrained model, Mar 2021.

Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. Backpropagation Applied to Handwritten Zip Code Recognition. *Neural Computation*, 1(4): 541–551, 12 1989. doi: 10.1162/neco.1989.1.4.541.

Chin-Yew Lin. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain, July 2004. Association for Computational Linguistics.

H. P. Luhn. The automatic creation of literature abstracts. *IBM Journal of Research and Development*, 2(2):159–165, 1958. doi: 10.1147/rd.22.0159.

Shuming Ma, Xu Sun, Jingjing Xu, Houfeng Wang, Wenjie Li, and Qi Su. Improving semantic relevance for sequence-to-sequence learning of chinese social media text summarization, 2017.

Ramesh Nallapati, Bowen Zhou, Cicero Nogueira dos santos, Caglar Gulcehre, and Bing Xiang. Abstractive text summarization using sequence-to-sequence rnns and beyond, 2016.

Xipeng Qiu, Jingjing Gong, and Xuanjing Huang. Overview of the nlpcc 2017 shared task: Chinese news headline categorization, 2017.

Alexander M. Rush, Sumit Chopra, and Jason Weston. A neural attention model for abstractive sentence summarization. In Lluís Màrquez, Chris Callison-Burch, and Jian Su, editors, *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 379–389, Lisbon, Portugal, September 2015. Association for Computational Linguistics. doi: 10.18653/v1/D15-1044.

M. Schuster and K.K. Paliwal. Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45(11):2673–2681, 1997. doi: 10.1109/78.650093.

Abigail See, Peter J. Liu, and Christopher D. Manning. Get to the point: Summarization with pointer-generator networks. In Regina Barzilay and Min-Yen Kan, editors, *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1073–1083, Vancouver, Canada, July 2017. Association for Computational Linguistics. doi: 10.18653/v1/P17-1099.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. Sequence to sequence learning with neural networks, 2014.

Zhaopeng Tu, Zhengdong Lu, Yang Liu, Xiaohua Liu, and Hang Li. Modeling coverage for neural machine translation. In Katrin Erk and Noah A. Smith, editors, *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 76–85, Berlin, Germany, August 2016. Association for Computational Linguistics. doi: 10.18653/v1/P16-1008.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2023.