

Transformer-based Lexically Constrained Headline Generation

Kosuke Yamada^{1*} Yuta Hitomi^{2†} Hideaki Tamori³ Ryohei Sasano¹
Naoki Okazaki⁴ Kentaro Inui^{5,6} Koichi Takeda¹

¹Nagoya University ²Insight Edge, Inc. ³The Asahi Shimbun Company

⁴Tokyo Institute of Technology ⁵Tohoku University ⁶RIKEN AIP

yamada.kosuke@c.mbox.nagoya-u.ac.jp,
yuta.hitomi@insightedge.jp, tamori-h@asahi.com,
{sasano,takedasu}@i.nagoya-u.ac.jp,
okazaki@c.titech.ac.jp, inui@ecei.tohoku.ac.jp

Abstract

This paper explores a variant of automatic headline generation methods, where a generated headline is required to include a given phrase such as a company or a product name. Previous methods using Transformer-based models generate a headline including a given phrase by providing the encoder with additional information corresponding to the given phrase. However, these methods cannot always include the phrase in the generated headline. Inspired by previous RNN-based methods generating token sequences in backward and forward directions from the given phrase, we propose a simple Transformer-based method that guarantees to include the given phrase in the high-quality generated headline. We also consider a new headline generation strategy that takes advantage of the controllable generation order of Transformer. Our experiments with the Japanese News Corpus demonstrate that our methods, which are guaranteed to include the phrase in the generated headline, achieve ROUGE scores comparable to previous Transformer-based methods. We also show that our generation strategy performs better than previous strategies.

1 Introduction

Following the initial work of Rush et al. (2015), abstractive headline generation using the encoder-decoder model has been studied extensively (Chopra et al., 2016; Nallapati et al., 2016; Paulus et al., 2018). In the automatic headline generation for advertising articles, there are requests to include a given phrase such as a company or product name in the headline.

Generating a headline that includes a given phrase has been considered one of the lexically constrained sentence generation tasks. For these tasks,

there are two major approaches. One approach is to select a plausible sentence including the given phrase from several candidate sentences generated from left to right (Hokamp and Liu, 2017; Anderson et al., 2017; Post and Vilar, 2018). Although these methods can include multiple phrases in a generated sentence, they are computationally expensive due to the large search space of the decoding process. In addition, since they try to force given phrases into sentences at every step of the generation process, these methods may harm the quality of the generated sentence (Liu et al., 2019).

Another approach proposed by Mou et al. (2015) is to generate token sequences in backward and forward directions from the given phrase. Mou et al. (2016) proposed Sequence to Backward and Forward Sequences (Seq2BF), which applies the method of Mou et al. (2015) to the sequence-to-sequence (seq2seq) framework. They use an RNN-based model and adopt the best strategies proposed by Mou et al. (2015), generating the backward sequence from the phrase and then generating the remaining forward sequence. Liu et al. (2019) introduced the Generative Adversarial Network (GAN) to the model of Mou et al. (2015) to resolve the exposure bias problem (Bengio et al., 2015) caused by generating sequences individually, and used the attention mechanism (Bahdanau et al., 2015) to improve the consistency between both sequences. However, their model does not support the seq2seq framework.

Recently, He et al. (2020) used a Transformer-based model (Vaswani et al., 2017), which is reported to achieve high performance, to generate a headline containing a given phrase. They proposed providing an encoder with additional information related to the given phrase. However, their method may not always include the given phrases in the generated headline.

In this study, we work on generating lexically constrained headlines using Transformer-based

*Work done during an internship at The Asahi Shimbun Company

†Work done at The Asahi Shimbun Company

Seq2BF. The RNN-based model used by Mou et al. (2016) executes a strategy of continuous generation in one direction, and thus cannot utilize the information of the forward sequence when generating the backward sequence. However, Transformer can execute a variety of generative strategies by devising attention masks, so it can solve the problem of the RNN-based model. We propose a new strategy that generates each token from a given phrase alternately in the backward and forward directions, in addition to adapting and extending the strategies of Mou et al. (2016) to the Transformer architecture.

Our experiments with a Japanese summarization corpus show that our proposed method always includes the given phrase in the generated headline and achieves performance comparable to previous Transformer-based methods. We also show that our proposed generating strategy performs better than the extended strategy of the previous methods.

2 Proposed Method

We propose a Transformer-based Seq2BF model that applies Seq2BF proposed by Mou et al. (2016) to the Transformer model to generate headlines including a given phrase. The Seq2BF takes $W (= w_1, \dots, w_L; w_{1:L})$ as the given phrase consisting of L tokens and generates the headline $y_{-M:-1}$ of M tokens backward from W , and the headline $y_{1:N}$ of N tokens forward from W . The Transformer-based Seq2BF is the Transformer model with two generation components, consisting of a linear and a softmax layer (see Figure 1).

In Transformer-based Seq2BF unlike Transformer generating tokens from left to right, the token position changes relatively depending on already generated tokens. We determine the token position, inputting to the positional encoding layer of the decoder, $\lfloor \frac{L+1}{2} \rfloor$ in W to be 0, and the position in the backward direction to be negative, and the position in the forward direction to be positive.

We consider the following four generation strategies. In addition to two strategies (a) and (b), which extend those proposed by Mou et al. (2016), we propose new strategies (c) and (d) as step-wise alternating generation to keep better contextual consistency in both backward and forward directions.

- (a) Generating a **sequence backward** and then a sequence forward. (**Seq-B**)
- (b) Generating a **sequence forward** and then a sequence backward. (**Seq-F**)

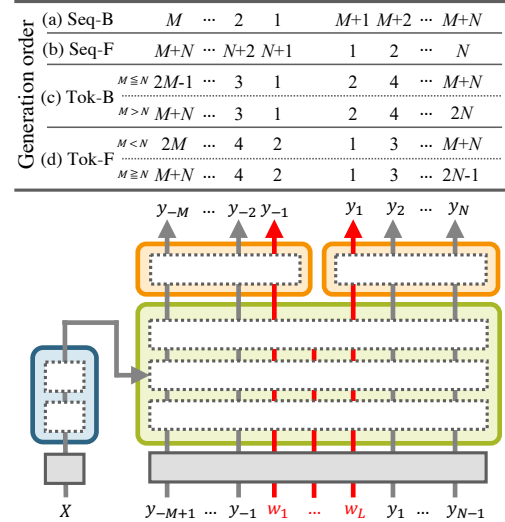


Figure 1: Overview of Transformer-based Seq2BF. Blue, green, and orange boxes indicate Transformer encoder, decoder, and generation components, respectively. The arrow from the encoder to the decoder represents that the decoder’s attention mechanism refers to the output from the encoder.

- (c) Generating each **token backward** and then forward alternately. (**Tok-B**)
- (d) Generating each **token forward** and then backward alternately. (**Tok-F**)

Transformer-based Seq2BF is formulated as

$$P(Y|X, W) = \prod_{j \in POS_j} P(y_j | Y_{obs}, X), \quad (1)$$

where X denotes tokens of the article, W denotes tokens of the given phrase, $Y (= y_{-M:-1}, w_{1:L}, y_{1:N})$ denotes tokens of the final generated headline, and Y_{obs} denotes the already-generated partial headline including W . Also, POS_j denotes a list of token positions representing the order of tokens to be generated corresponding to each generation strategy (see Figure 1), for example $[-1, -2, \dots, -M, 1, 2, \dots, N]$ in Seq-B. In Tok-B/F which M and N are different, once the generation in one direction is completed, the generation will be continued only in the remaining directions until $M + N$ steps. For example in the case of $M > N$ in Tok-B, our method completes generating tokens in the forward direction first, so it generates them in both directions until the $2N$ step, and then generates them only in the backward direction from the $2N + 1$ step to the $M + N$ step.

To train the model on these generative strategies, we have prepared an attention mask for the

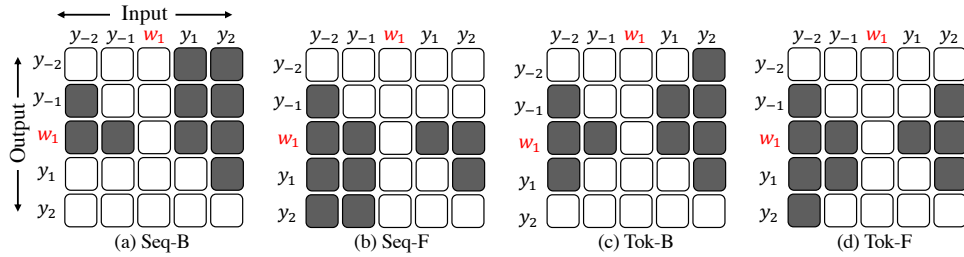


Figure 2: Attention mask patterns on the decoder side during training, used for each generation strategy in Transformer-based Seq2BF. The dark cells indicate the masked areas. These are examples of the headline with the length of 5, where w_1 is specified as the phrase.

decoder. Transformer can control the generation order of tokens by devising the attention mask used in the decoder’s self-attention mechanism. Transformer generates tokens from left to right, so it is sufficient to disable the attention to tokens forward from the input tokens. However, the Transformer-based Seq2BF needs to specify the areas where input tokens disallow the attention in the backward and forward directions, depending on each generation strategy (see Figure 2).

3 Experiment

We conducted the experiment to verify the performance of our methods in the headline generation task. The objective of our experiment is to compare our method with previous Transformer-based methods that generate tokens from left to right. We also compare Seq-B/F, the generation orders proposed by Mou et al. (2016), with Tok-B/F, our new generation orders.

3.1 Setting

We used the 2019 version of the Japanese News Corpus (JNC)¹ (Hitomi et al., 2019) as the dataset. The JNC contains 1,932,399 article-headline pairs, and we split them randomly at a ratio of 98:1:1 for use as training, validation, and test sets, respectively.² We utilized MeCab (Kudo et al., 2004) with the IPAdic³ and then applied the Byte Pair Encoding (BPE) algorithm⁴ (Gage, 1994) for tokenization. We trained BPE with 10,000 merge operations and obtained the most frequent 32,000

tokens from the articles and the headlines, respectively.

We used context word sequences extracted from the reference headlines by GiNZA⁵ as the ‘given’ phrase.⁶ An average of 4.99 phrases was extracted from the reference headlines, and the ‘given’ phrases consisted of an average of 2.32 tokens. We evaluated our methods using precision, recall, and F-score of ROUGE-1/2/L (Lin, 2004) and success rate (SR), which is the percentage of the headline that includes the given phrase. We also calculated the Average Length Difference (ALD) to analyze the length of the generated headlines, as

$$\text{ALD} = \frac{1}{n} \sum_{i=1}^n l_i - \text{len}_i, \quad (2)$$

where n , l_i , and len_i are the number of samples, the length of the generated headline, and the length of the reference headline, respectively.

As a comparison method, we adopted the method proposed by He et al. (2020) with vanilla Transformer instead of BART (Lewis et al., 2020). This method controls the output by inserting the given phrase and the special token ‘ \prime ’ in front of the input articles and randomly drops the given phrase from the input articles during training to improve the performance. The hyperparameters of both the comparison and our models are determined as described in Vaswani et al. (2017). The training was terminated when the perplexity computed on the validation set did not update three times in a row, and we used the model with the minimum perplexity on the validation set. The beam size during the inference was set to three.

¹https://cl.asahi.com/api_data/jnc-jamul-en.html

²We applied the preprocessing script at <https://github.com/asahi-research/script-for-transformer-based-seq2bf> to the original JNC to obtain the split dataset.

³<https://taku910.github.io/mecab/>

⁴<https://github.com/rsennrich/subword-nmt>

⁵<https://github.com/megagonlabs/ginza>

⁶We used “ginza.bunsetu_phrase_spans” API.

	SR	ROUGE-1 P/R/F	ROUGE-2 P/R/F	ROUGE-L P/R/F	ALD	params $\times 10^6$	
Transformer (Vaswani et al., 2017)	36.3	57.1/48.9/51.4	29.8/25.2/26.5	47.1/40.9/42.8	-3.62	72	
Transformer version of He et al. (2020)	90.2	63.1/ 54.8 /57.2	36.0/30.7/32.2	51.9/ 45.4 /47.4	-3.02	72	
Transformer-based Seq2BF	(Seq-B)	100.0	63.6/52.4/55.8	37.4/30.2/32.3	54.3/44.2/47.5	-4.19	80
	(Seq-F)	100.0	64.6/53.2/56.7	38.1/ 30.8 /32.9	54.8/44.9/48.1	-4.30	80
	(Tok-B)	100.0	66.6/52.9/ 57.6	39.3/ 30.8 / 33.6	55.6/45.0/ 48.6	-5.29	80
	(Tok-F)	100.0	67.6 /51.6/57.1	40.1 /30.2/33.5	56.7 /44.2/ 48.6	-6.05	80

Table 1: Experimental results. SR means success rate, and P/R/F means Precision/Recall/F-score.

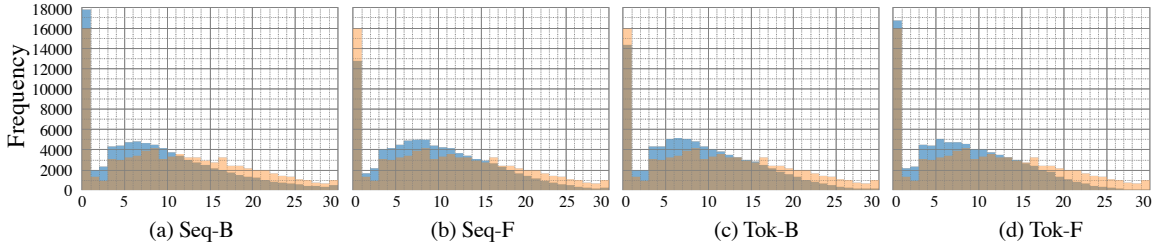


Figure 3: Histogram of the character-level position of the given phrase in the headlines generated by Transformer-based Seq2BF. Blue and orange bars indicate the generated and reference headlines, respectively.

3.2 Results

Table 1 shows the experimental results. Note that the proposed and compared methods achieved higher ROUGE scores than Transformer because we computed ROUGE scores between the reference and the system-generated headlines, including the phrase extracted from the reference headlines.

Our methods always include the given phrase in the generated headlines, whereas the comparison method had a success rate of around 90%. Although the recall of ROUGE scores tended to be higher in the comparison method than in the proposed method, the precision and F-scores of ROUGE scores in the proposed method were comparable or higher than in the comparison method. As we notice from ALD, we found that Transformer-based Seq2BF generated shorter headlines than the Transformer models. It has been confirmed that the Transformer models with a single output direction tend to generate shorter headlines than the reference. Because Transformer-based Seq2BF has two output directions, the generated headlines were considered to be even shorter. This is the reason why our methods had lower recall scores than the comparison methods. Comparing the generation strategies of Transformer-based Seq2BF, we can see that Tok-B/F had a higher score than Seq-B/F.

To analyze how the four generation strategies of Transformer-based Seq2BF affected the system-generated headlines, we showed the character-level

position of the given phrase in the headline using histograms in Figure 3. As we can see, all generation strategies had similar distributions in the reference and system-generated headlines, and hence Transformer-based Seq2BF has also been presumed to learn the position of a given phrase in the headline. Focusing on the headlines that include the given phrase in the head, the difference between the reference and the headline generated by Tok-B/F is smaller than that of the headline generated by Seq-B/F. Also, the headlines generated by Seq-B tend to place the given phrase in the beginning, while this tendency is opposite for the headlines generated by Seq-F.

Table 2 shows examples of the headlines generated by the Transformer-based Seq2BF (Tok-B). When a product name such as “桜とイワシのパフェ” (“Cherry Blossom and Sardine Parfait”) was given, our methods could generate a natural headline that includes the given phrase. Also, given the phrase “6月末” (“the End of June”), our methods generated a headline with the addition of “販売” (“on Sale”) that matched the given phrase. On the other hand, we found the problem of generating the same words related to the given phrase in the backward and forward directions, such as the headline generated given “群れ” (“Schools”). In addition, given the phrase “約1万匹” (“About 10,000”), our methods generated the headline meaning that special sweets contain about 10,000 sardines. In this way, examples that were not faithful to the article were confirmed.

Article: 約1万匹のイワシが群れで泳ぐ様子を見られる京都水族館。この展示に合わせ、ちょっぴり変わった特別スイーツが6月末まで販売される。名前は「桜といわしのパフェ」で、...

At the Kyoto Aquarium, you can see about 10,000 sardines swimming in schools. To coincide with this exhibition, a special sweet that is slightly unique will be on sale until the end of June. It is called “Cherry Blossom and Sardine Parfait,” and ...

Reference Headline: 「目からウロコのおいしさ」 京都水族館にイワシパフェ
“Scales Falling from Your Eyes” – Kyoto Aquarium Serves Sardine Parfait

Given Phrases	Generated Headlines
桜といわしのパフェ Cherry Blossom and Sardine Parfait	「桜といわしのパフェ」特別スイーツ 京都水族館 Special Sweets “Cherry Blossom and Sardine Parfait” at Kyoto Aquarium
6月末 the End of June	イワシの特別スイーツ、6月末まで販売 京都水族館 Special Sardine Sweets on Sale at Kyoto Aquarium until the End of June
群れ Schools	イワシの群れのイワシ、特別スイーツに 京都水族館 Sardines of Schools of Sardines, to Special Sweets at Kyoto Aquarium
約1万匹 About 10,000	イワシ約1万匹の特別スイーツ 京都水族館 Special Sweets of About 10,000 Sardines at Kyoto Aquarium

Table 2: Examples of headlines generated by Transformer-based Seq2BF (Tok-B).

As can be seen from Table 2, various headlines are generated according to given phrases. In general, it is difficult to control the diversity in headline generation, but our methods can generate diverse headlines by giving a variety of phrases. However, it may be necessary to discuss whether our methods could generate diverse headlines. The reason is that all examples are only partially diverse. Specifically, they always include “特別スイーツ” (“Special Sweets”) and “京都水族館” (“Kyoto Aquarium”) as important contents in the headline.

4 Conclusion

We proposed Transformer-based Seq2BF that generates the lexically constrained headline by devising the attention mask for the decoder and generating backward and forward sentences from the phrase. Our experiments using the JNC demonstrated that Transformer-based Seq2BF always includes the given phrase in the generated headline and obtains comparable performance compared to previous Transformer-based methods. We also showed that strategies of generating each token alternately between backward and forward directions are more effective than that of generating a sequence in one direction and then a sequence in another direction.

In future work, we will investigate whether Transformer-based Seq2BF can generate natural headlines even when given a variety of phrases, such as phrases not in the reference or the article, and examine if our methods can creatively generate diverse headlines by giving a variety of phrases quantitatively. Also, we will explore methods for generating headlines that include multiple phrases.

References

- Peter Anderson, Basura Fernando, Mark Johnson, and Stephen Gould. 2017. Guided open vocabulary image captioning with constrained beam search. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (EMNLP 2017)*, pages 936–945.
- Dzmitry Bahdanau, Kyung Hyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *Proceedings of the 3rd International Conference on Learning Representations (ICLR 2015)*.
- Samy Bengio, Oriol Vinyals, Navdeep Jaitly, and Noam Shazeer. 2015. Scheduled sampling for sequence prediction with recurrent neural networks. In *Proceedings of the 28th International Conference on Neural Information Processing (NIPS 2015)*, pages 1171–1179.
- Sumit Chopra, Michael Auli, and Alexander M. Rush. 2016. Abstractive sentence summarization with attentive recurrent neural networks. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT 2016)*, pages 93–98.
- Philip Gage. 1994. A new algorithm for data compression. *C Users Journal*, 12(2):23–38.
- Junxian He, Wojciech Kryściński, Bryan McCann, Nazneen Rajani, and Caiming Xiong. 2020. CTRL-sum: Towards generic controllable text summarization. *arXiv preprint arXiv:2012.04281*.
- Yuta Hitomi, Yuya Taguchi, Hideaki Tamori, Ko Kikuta, Jiro Nishitoba, Naoaki Okazaki, Kentaro Inui, and Manabu Okumura. 2019. A large-scale multi-length headline corpus for analyzing length-constrained headline generation model evaluation. In *Proceedings of the 12th International Conference on Natural Language Generation (INLG 2019)*, pages 333–343.

- Chris Hokamp and Qun Liu. 2017. Lexically constrained decoding for sequence generation using grid beam search. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL 2017)*, pages 1535–1546.
- Taku Kudo, Kaoru Yamamoto, and Yuji Matsumoto. 2004. Applying conditional random fields to Japanese morphological analysis. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing (EMNLP 2004)*, pages 230–237.
- Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL 2020)*, pages 7871–7880.
- Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Proceedings of the Workshop on Text Summarization Branches Out*, pages 74–81.
- Dayiheng Liu, Jie Fu, Qian Qu, and Jiancheng Lv. 2019. BFGAN: Backward and forward generative adversarial networks for lexically constrained sentence generation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 27(12):2350–2361.
- Lili Mou, Yiping Song, Rui Yan, Ge Li, Lu Zhang, and Zhi Jin. 2016. Sequence to backward and forward sequences: A content-introducing approach to generative short-text conversation. In *Proceedings of the 26th International Conference on Computational Linguistics: Technical Papers (COLING 2016)*, pages 3349–3358.
- Lili Mou, Rui Yan, Ge Li, Lu Zhang, and Zhi Jin. 2015. Backward and forward language modeling for constrained sentence generation. *arXiv preprint arXiv:1512.06612*.
- Ramesh Nallapati, Bowen Zhou, Cicero dos Santos, Caglar Gulcehre, and Bing Xiang. 2016. Abstractive text summarization using sequence-to-sequence RNNs and beyond. In *Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning (CoNLL 2016)*, pages 280–290.
- Romain Paulus, Caiming Xiong, and Richard Socher. 2018. A deep reinforced model for abstractive summarization. In *Proceedings of the 6th International Conference on Learning Representations (ICLR 2018)*.
- Matt Post and David Vilar. 2018. Fast lexically constrained decoding with dynamic beam allocation for neural machine translation. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT 2018)*, pages 1314–1324.
- Alexander M. Rush, Sumit Chopra, and Jason Weston. 2015. A neural attention model for abstractive sentence summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP 2015)*, pages 379–389.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems (NIPS 2017)*, pages 5998–6008.