R-TeaFor: Regularized Teacher-Forcing for Abstractive Summarization

Guan-Yu Lin
National Taiwan University
r09944017@ntu.edu.tw

Pu-Jen Cheng
National Taiwan University
pjcheng@csie.ntu.edu.tw

Abstract

Teacher-forcing is widely used in training sequence generation models to improve sampling efficiency and to stabilize training. However, teacher-forcing is vulnerable to the exposure bias problem. Previous works have attempted to address exposure bias by modifying the training data to simulate model-generated results. Nevertheless, they do not consider the pairwise relationship between the original training data and the modified ones, which provides more information during training. Hence, we propose Regularized Teacher-Forcing (R-TeaFor) to utilize this relationship for better regularization. Empirically, our experiments show that R-TeaFor outperforms previous summarization state-of-the-art models, and the results can be generalized to different pre-trained models.

1 Introduction

Recently, the encoder-decoder models have made a giant leap in sequence generation tasks. These models are primarily trained with the teacher-forcing method (Goodfellow et al., 2016), which has been praised for enhancing sampling efficiency and training stability. However, as teacher-forcing only exposes models to the training data distribution rather than their prediction distribution, models suffer from the exposure bias problem (Bengio et al., 2015; Ranzato et al., 2015).

Previous works for solving exposure bias can be roughly split into four categories: regularization, contrastive learning, reinforcement learning, and sampling-based methods. Regularization methods such as dropout (Srivastava et al., 2014) and perturbation (Goodfellow et al., 2015) focus on problems of data distribution differences between training data and inference data, which is also related to exposure bias. Liang et al. (2021) propose R-Drop, which refines dropout by forcing predictions of different sub-models generated by dropout to be consistent with each other. Gal and Ghahramani (2016) apply token-level word dropout in language modeling. Takase and Kiyono (2021) then introduce word dropout to sequence generation. Other token-level perturbations include replacing tokens with source tokens, model-generated tokens (Bengio et al., 2015), and a mixture of them (Zhang et al., 2019). Adversarial perturbation (Sato et al., 2019; Aghajanyan et al., 2021) is another alternative that directly modifies the token embedding. Although widely used, most regularization methods are not exclusively designed for the exposure bias problem. Hence, their ability to alleviate exposure bias is limited.

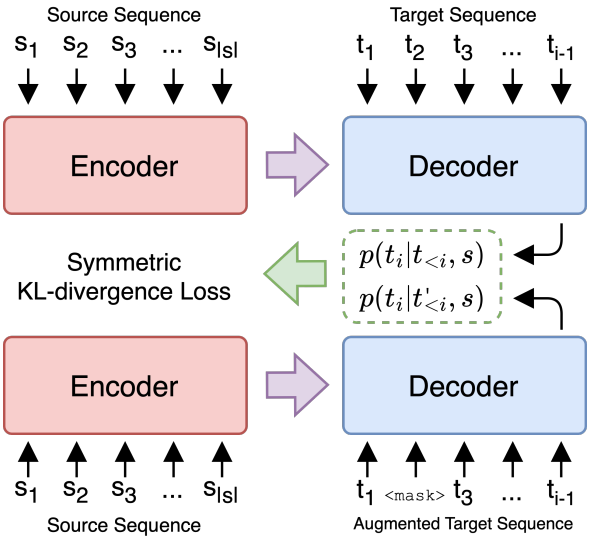

Figure 1: An illustration of R-TeaFor. The output distributions are generated both by the original target sequence and the augmented target sequence. R-TeaFor aims to reduce the distance of two distributions by additional symmetric KL-divergence loss.

Contrastive learning is widely used in either pre-training or fine-tuning sequence generation models to provide more information inside the training data. Word vectors (Mikolov et al., 2013) and language models are the early applications for contrastive learning in natural language processing.

Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, pages 6303 - 6311
December 7-11, 2022

For summarization tasks, both extractive-based approach (Zhong et al., 2020) and abstractive-based approach can be combined with contrastive learning. These methods mostly entail an additional contrastive loss (Lee et al., 2021; Pan et al., 2021; Xu et al., 2021), or use the contrastive data to learn an additional model for reranking (Liu and Liu, 2021). The challenge of contrastive learning is to find a suitable contrastive pair that is neither too easy to distinguish nor too hard to learn.

Reinforcement learning based on policy gradient (Sutton et al., 1999) and their variations have been used extensively for sequence generation (Ranzato et al., 2015; Pasunuru and Bansal, 2018; Paulus et al., 2018). Pang and He (2021) formulate the sequence generation learning problem as an off-line reinforcement learning with expert demonstrations that addresses exposure bias by training the model on its state/history distribution. However, reinforcement learning still requires a warm start phase or regularization powered by traditional teacher-forcing.

Scheduled sampling (Bengio et al., 2015) is a representative sampling-based approach that schedules the sampling rate between ground-truth tokens and model-generated tokens according to training steps. Zhang et al. (2019) propose another variation by refining the sampling rate with beam search. Liu et al. (2021) further suggest scheduling the sampling rate according to the decoding steps. Although sampling-based approaches can simulate the data distribution in the inference stage during training, the modified training data might deviate from the original one. Such deviation will mislead the model to learn from noisy training data.

We, therefore, present Regularized Teacher-Forcing (R-TeaFor), which considers the pairwise relationship between the original and modified data. Figure 1 illustrates the general framework of R-TeaFor. During training, the model generates the token distribution twice—once with the ground-truth sequence as input and once with the augmented sequence modified from the same ground-truth. R-TeaFor forces the above two distributions to be consistent with each other by introducing the symmetric KL-divergence loss. The augmented sequence serves as a regularizer to prevent the model from exposure bias. Meanwhile, the model can still learn from the original ground-truth sequences.

Though the concept of R-TeaFor is straightforward, we find it surprisingly effective through extensive experiments on two summarization benchmarks. Moreover, the results can be generalized to different pre-trained models such as BART (Lewis et al., 2020) and PEGASUS (Zhang et al., 2020).

2 R-TeaFor

Before elaborating on R-TeaFor, we first formulate the abstractive summarization problem in a more general sequence generation manner.

Sequence generation models take source sequence $s = s_1, s_2, ..., s_{|s|}$ as input, and generate target sequence $t = t_1, t_2, ..., t_{|t|}$ autoregressively. In other words, the generating process can be modeled by the following probability:

$$p(t|s) = \prod_{i=1}^{|t|} p(t_i|t_{<i}, s) \tag{1}$$

where $t_{<i} = t_1, t_2, ..., t_{i-1}$.

During training, the previous target sequence $t_{<i}$ consists of ground-truth tokens. However, as the ground-truth tokens become unavailable during inference, we apply model predictions in $t_{<i}$. This discrepancy leads to the exposure bias problem. R-TeaFor handles exposure bias by forcing the model to learn from both ground-truth sequence t and augmented sequence t' simultaneously.

For constructing the augmented sequence $t' = t'_1, t'_2, ..., t'_{|t'|}$, we randomly mask the ground-truth tokens t_i with probability β.

$$t'_i = \begin{cases} t_i & \text{with probability } 1-\beta \\ \texttt{<mask>} & \text{with probability } \beta \end{cases} \tag{2}$$

While decoding the i-th token, R-TeaFor generates the probability distribution of t_i twice—once with the ground-truth sequence and once with the augmented sequence. The basic negative log-likelihood loss is calculated as

$$\mathcal{L}^i_{nll} = \frac{1}{2}\big(-\log p(t_i|t_{<i}, s) - \log p(t_i|t'_{<i}, s)\big). \tag{3}$$

To further constrain the model's output, we introduce the symmetric KL-divergence in the loss function, which is

$$\begin{aligned}\mathcal{L}^i_{kld} = \frac{1}{2}\Big(&\mathcal{D}_{KL}\big(p(t_i|t_{<i}, s)||p(t_i|t'_{<i}, s)\big) \\ &+ \mathcal{D}_{KL}\big(p(t_i|t'_{<i}, s)||p(t_i|t_{<i}, s)\big)\Big)\end{aligned}. \tag{4}$$

Finally, we combine negative log-likelihood loss and symmetric KL-divergence loss with hyperparameter α controlling the weight of two losses.

$$\mathcal{L}^i = \mathcal{L}^i_{nll} + \alpha\mathcal{L}^i_{kld} \tag{5}$$

By this measure, R-TeaFor encourages the model to make a consistent prediction regardless of the input sequence being augmented or not. Unlike sampling-based methods which use original training data only in the warm-up phase, R-TeaFor always trains the model with pairwise original/augmented data. During training, the distribution generated based on the ground-truth sequence can serve as a guide when the model attempts to regularize on a much noisy augmented sequence. Without such guidance, models may regularize to a sub-optimal point as models are bounded with fewer constraints.

3 Experiments

3.1 Experimental Setup

Datasets and Metrics. We have conducted extensive experiments on CNN/DailyMail (Hermann et al., 2015) and XSum (Narayan et al., 2018) benchmarks. Appendix A summarizes the statistics of the datasets. For evaluation, we use ROUGE f-measures (Lin, 2004).

Training Setup. The pre-trained models of BART and PEGASUS are used for fine-tuning. We set $\alpha = 0.04$ and $\beta = 0.4$, and apply the label smoothing cross-entropy (Pereyra et al., 2017). Other detailed fine-tuning parameters mostly follow the original papers of BART and PEGASUS. Appendix B provides the complete setup.

Generation Setup. The beam size is set to 3 in CNN/DailyMail and 5 in XSum. Duplicated trigrams have been removed in beam search. We do a minimal parameter search over min-len, max-len, and length penalty on the validation set (Fan et al., 2018). The final results are given in Appendix B.

3.2 Main Results

The results of R-TeaFor on different summarization benchmarks are shown in Table 1. Notably, R-TeaFor outperforms the previous state-of-the-art models with both BART and PEGASUS, indicating that the application of R-TeaFor is not restricted to the specific pre-trained model. For the sake of completeness, we also attach the effect of setting different α and β during training in Appendix C.

3.3 Two Augmentation Rates

In this section, we examine the training data of R-TeaFor. The pairwise training data consists of sequences with 0.0 probability augmented on one side and 1.0 probability on another. We vary the probability (i.e., the augmentation rate) on both sides to discuss the influence of augmentation. For augmented sequences, we follow the rules described in Eq. 2.

Model	R1 / R2 / RL
CNN/DailyMail	
ProphetNet (Qi et al., 2020)	43.68 / 20.64 / 40.72
BART+R3F (Aghajanyan et al., 2021)	44.38 / 21.53 / 41.17
SS-decoding (Liu et al., 2021)	44.40 / 21.44 / 41.61
GOLD-p (Pang and He, 2021)	45.40 / 22.01 / 42.25
GOLD-s (Pang and He, 2021)	44.82 / 22.09 / 41.81
SeqCo (Xu et al., 2021)	45.02 / 21.80 / 41.75
GSum (Dou et al., 2021)	**45.94** / 22.32 / 42.48
$\text{BART}_{\text{large}}$ (Lewis et al., 2020)	44.16 / 21.28 / 40.90
+ R-TeaFor	45.52 / **22.62** / **42.71**
$\text{PEGASUS}_{\text{large}}$ (Zhang et al., 2020)	44.17 / 21.47 / 41.11
+ R-TeaFor	44.72 / 22.12 / 41.82
XSum	
GOLD-p (Pang and He, 2021)	45.75 / 22.26 / 37.30
GOLD-s (Pang and He, 2021)	45.85 / 22.58 / 37.65
SeqCo (Xu et al., 2021)	45.65 / 22.41 / 37.04
GSum (Dou et al., 2021)	45.40 / 21.89 / 36.67
$\text{BART}_{\text{large}}$ (Lewis et al., 2020)	45.14 / 22.27 / 37.25
+ R-TeaFor	46.14 / 22.43 / 37.69
$\text{PEGASUS}_{\text{large}}$ (Zhang et al., 2020)	47.21 / 24.56 / 39.25
+ R-TeaFor	**47.90** / **24.59** / **39.64**

Table 1: Results on standard summarization benchmarks. R1/R2/RL represents ROUGE-1/ROUGE-2/ROUGE-L, respectively. For other models, we report the results in the original papers. ProphetNet, BART+R3F, and SS-decoding do not include XSum results in their papers. The highest numbers are in bold.

By looking at Figure 2, we can conclude three observations. First of all, no augmentation on both sides (0.0, 0.0) reduces the model to regularizing on two dropout results (Liang et al., 2021), which is inferior to models applying any pair of augmentation rates. Secondly, full augmentation on both sides (1.0, 1.0) transforms the model into regularizing on two word dropout results (Takase and Kiyono, 2021). The model has been regularized stronger as more perturbation is introduced. Lastly, the R-TeaFor setting (0.0, 1,0) performs best among all augmentation rate combinations.

The experimental result implies that augmentation is only required on one side of pairwise training sequences, and any additional augmentation above is unnecessary. Furthermore, maximizing the pairwise relationship of original/augmented data (0.0, 1.0) results in the best regularization compared with other settings having the same degree of augmentation (i.e., (0.2, 0.8), (0.4, 0.6)).

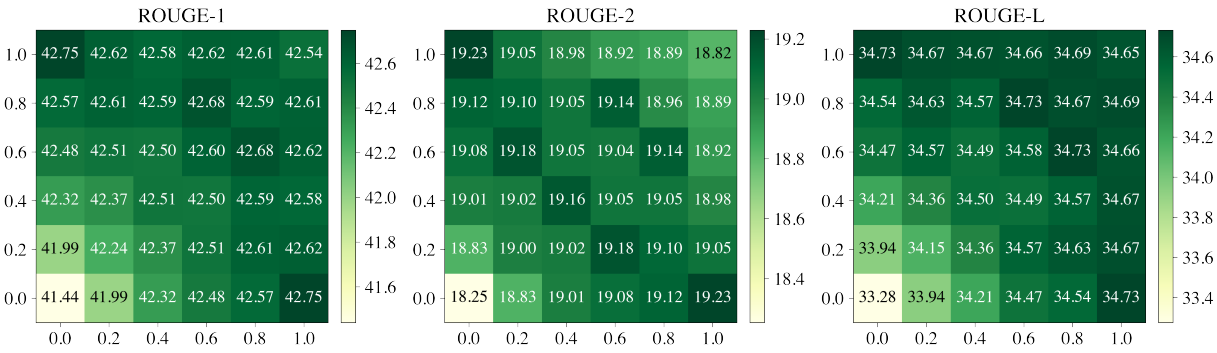

Figure 2: ROUGE scores of $\text{BART}_{\text{base}}$ on XSum when different augmentation rates are applied to each side of pairwise training sequences. The augmentation rate is the ratio of sequences being augmented in the training data. For the augmented sequences, we follow the setting of augmenting with mask token described in Eq. 2, and set $\beta = 0.4$. The tables are symmetric and triangular since the combination of two augmentation rates is commutative.

3.4 Augmentation Strategies

We compare different augmentation strategies by fixing one side of the sequence pairs as ground-truth sequence and changing the augmentation strategy of another. The augmentation strategies mainly follow Eq. 2, but we modify the mask token to the model-generated token or random token. Table 2 compares augmentation strategies with two baselines, the original teacher-forcing and training with ground-truth sequence pairs (i.e., the (0.0, 0.0) setting described in Section 3.3). Empirically, augmenting with mask token results in the best performance. Also, training with only the ground-truth sequence pairs will not significantly improve the performance, which echoes with Section 3.3.

Augmentation Strategies	R1	R2	RL
Teacher-Forcing	41.37	18.20	33.13
Ground-Truth Pairs	41.44	18.25	33.28
Model-Generated	42.42	18.83	34.46
Random	42.64	19.08	34.69
Mask	42.72	19.24	34.75

Table 2: ROUGE scores of $\text{BART}_{\text{base}}$ with different augmentation strategies on XSum. R1/R2/RL represents ROUGE-1/ROUGE-2/ROUGE-L, respectively.

3.5 Prediction Similarity

We want to examine the ability of R-TeaFor to mitigate exposure bias. Each model receives two inputs during this experiment: ground-truth sequence and model-generated sequence. The token prediction distributions generated from two inputs are compared for cosine similarity calculation. We examine four models: BART, BART + Pairwise, BART + Augmentation, and BART + R-TeaFor. The BART and BART + Augmentation are both trained with traditional teacher-forcing where BART + Augmentation has the same additional augmentation data as BART + R-TeaFor. The BART + Pairwise model is trained with pairwise ground-truth sequences (i.e., the (0.0, 0.0) setting described in Section 3.3).

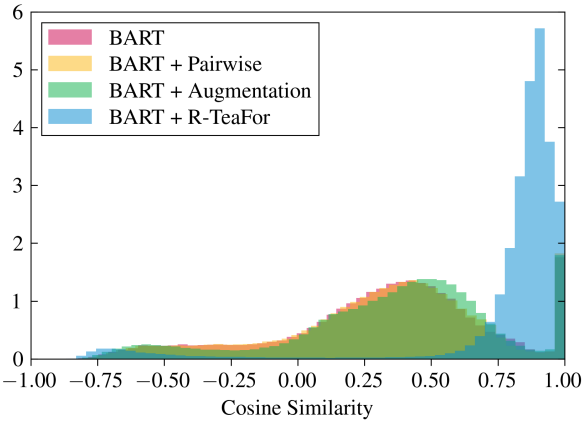

Figure 3: Cosine similarity of different models tested on XSum benchmark. For each model, We calculate the cosine similarity between their decoder outputs from ground-truth sequence inputs and model-generated ones. As the figure suggests, 85.5% of the BART + R-TeaFor predictions have a cosine similarity higher than 0.75. Hence, we can infer that R-TeaFor leads to a more consistent output across training/inference.

Figure 3 shows the probability density of different models. BART + Pairwise barely improves the similarity of token prediction distributions as it only addresses the randomness introduced by dropout. BART + Augmentation has marginal improvement due to the additional augmented data. Compared with BART + Augmentation, BART + R-TeaFor has an additional pairwise relationship

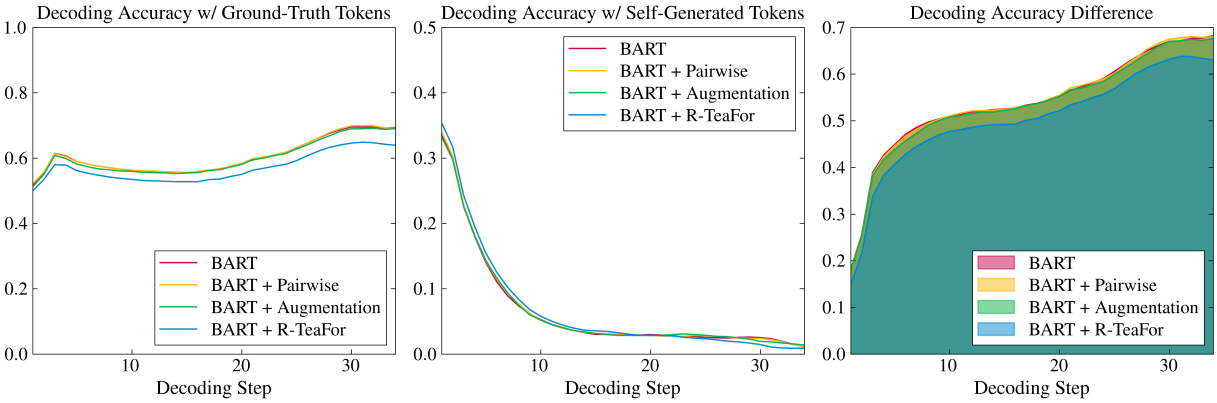

Figure 4: Decoding accuracy at each decoding step on XSum while the decoder receives different input sequences. Although the original BART performs better on ground-truth token inputs, a conventional teacher-forcing setting, the performance degrades when self-generated tokens are used as decoder inputs.

between ground-truth sequences and augmented ones, rendering regularizing different inputs easier. R-TeaFor significantly improves the similarity as over 85.5% of predictions have cosine similarity larger than 0.75. Hence, R-TeaFor has a more consistent output across ground-truth inputs and model-generated inputs, which demonstrates that R-TeaFor can mitigate exposure bias.

3.6 Decoding Accuracy

Figure 4 demonstrates the decoding accuracy of models receiving different previous target sequence $t_{<i}$ as input during inference. When models receive ground-truth tokens as $t_{<i}$, the decoding accuracy remains approximately the same. However, when models receive self-generated tokens as $t_{<i}$, the accuracy drops significantly as the decoding error accumulates with the growth of the decoding steps. Despite the performance drop, R-TeaFor maintains a higher decoding accuracy compared with other methods. Moreover, R-TeaFor has less decoder accuracy difference between receiving ground-truth inputs and self-generated inputs. This again proves that R-TeaFor can mitigate exposure bias.

4 Conclusion

We present Regularized Teacher-Forcing (R-TeaFor), which utilizes the pairwise relationship between the original training data and augmented data for model training. The original data can serve as a guide when the model attempts to regularize on much noisy augmented data. With extensive experiments, we discuss the necessity of the pairwise relationship in R-TeaFor. In addition, we show that R-TeaFor outperforms previous state-of-the-art models on CNN/DailyMail and XSum benchmarks. We believe that R-TeaFor has strong potential to apply to other sequence generation tasks such as machine translation and question answering.

Limitations

R-TeaFor has a more substantial regularization effect in the earlier decoding steps, and we find it performs better when summarizing longer text. R-TeaFor aims to tackle the exposure bias problem in summarization or potentially other sequence generation tasks. The effectiveness of R-TeaFor is limited to the improvement of exposure bias.

Like other deep learning models, the training time for R-TeaFor is more acceptable when training with GPUs. In our experiments, we use 1 Tesla V100 for all the training and inference. The training time of one single run is 48 hr.

References

Armen Aghajanyan, Akshat Shrivastava, Anchit Gupta, Naman Goyal, Luke Zettlemoyer, and Sonal Gupta. 2021. Better fine-tuning by reducing representational collapse. In *International Conference on Learning Representations*.

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 Systems - Volume 1*, NIPS'15, page 1171–1179, Cambridge, MA, USA. MIT Press.

Zi-Yi Dou, Pengfei Liu, Hiroaki Hayashi, Zhengbao Jiang, and Graham Neubig. 2021. GSum: A general framework for guided neural abstractive summarization. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for*

Computational Linguistics: Human Language Technologies, pages 4830–4842, Online. Association for Computational Linguistics.

Angela Fan, David Grangier, and Michael Auli. 2018. Controllable abstractive summarization. In *Proceedings of the 2nd Workshop on Neural Machine Translation and Generation*, pages 45–54, Melbourne, Australia. Association for Computational Linguistics.

Yarin Gal and Zoubin Ghahramani. 2016. A theoretically grounded application of dropout in recurrent neural networks. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, NIPS'16, page 1027–1035, Red Hook, NY, USA. Curran Associates Inc.

Ian Goodfellow, Yoshua Bengio, and Aaron Courville. 2016. *Deep Learning*. MIT Press. http://www.deeplearningbook.org.

Ian Goodfellow, Jonathon Shlens, and Christian Szegedy. 2015. Explaining and harnessing adversarial examples. In *International Conference on Learning Representations*.

Karl Moritz Hermann, Tomáš Kočiský, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'15, page 1693–1701, Cambridge, MA, USA. MIT Press.

Seanie Lee, Dong Bok Lee, and Sung Ju Hwang. 2021. Contrastive learning with adversarial perturbations for conditional text generation. In *International Conference on Learning Representations*.

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*, pages 7871–7880, Online. Association for Computational Linguistics.

Xiaobo Liang, Lijun Wu, Juntao Li, Yue Wang, Qi Meng, Tao Qin, Wei Chen, Min Zhang, and Tie-Yan Liu. 2021. R-drop: Regularized dropout for neural networks. In *Advances in Neural Information Processing Systems*.

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

Yijin Liu, Fandong Meng, Yufeng Chen, Jinan Xu, and Jie Zhou. 2021. Scheduled sampling based on decoding steps for neural machine translation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3285–3296, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Yixin Liu and Pengfei Liu. 2021. SimCLS: A simple framework for contrastive learning of abstractive summarization. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 1065–1072, Online. Association for Computational Linguistics.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space.

Shashi Narayan, Shay B. Cohen, and Mirella Lapata. 2018. Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1797–1807, Brussels, Belgium. Association for Computational Linguistics.

Xiao Pan, Mingxuan Wang, Liwei Wu, and Lei Li. 2021. Contrastive learning for many-to-many multilingual neural machine translation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 244–258, Online. Association for Computational Linguistics.

Richard Yuanzhe Pang and He He. 2021. Text generation by learning from demonstrations. In *International Conference on Learning Representations*.

Ramakanth Pasunuru and Mohit Bansal. 2018. Multi-reward reinforced summarization with saliency and entailment. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 646–653, New Orleans, Louisiana. Association for Computational Linguistics.

Romain Paulus, Caiming Xiong, and Richard Socher. 2018. A deep reinforced model for abstractive summarization. In *International Conference on Learning Representations*.

Gabriel Pereyra, George Tucker, Jan Chorowski, Lukasz Kaiser, and Geoffrey Hinton. 2017. Regularizing neural networks by penalizing confident output distributions. In *International Conference on Learning Representations*.

Weizhen Qi, Yu Yan, Yeyun Gong, Dayiheng Liu, Nan Duan, Jiusheng Chen, Ruofei Zhang, and Ming Zhou. 2020. ProphetNet: Predicting future n-gram for sequence-to-SequencePre-training. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2401–2410, Online. Association for Computational Linguistics.

Marc'Aurelio Ranzato, Sumit Chopra, Michael Auli, and Wojciech Zaremba. 2015. Sequence level training with recurrent neural networks.

Motoki Sato, Jun Suzuki, and Shun Kiyono. 2019. Effective adversarial regularization for neural machine translation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 204–210, Florence, Italy. Association for Computational Linguistics.

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1073–1083, Vancouver, Canada. Association for Computational Linguistics.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: A simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.*, 15(1):1929–1958.

Richard S. Sutton, David McAllester, Satinder Singh, and Yishay Mansour. 1999. Policy gradient methods for reinforcement learning with function approximation. In *Proceedings of the 12th International Conference on Neural Information Processing Systems*, NIPS'99, page 1057–1063, Cambridge, MA, USA. MIT Press.

Sho Takase and Shun Kiyono. 2021. Rethinking perturbations in encoder-decoders for fast training. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5767–5780, Online. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Shusheng Xu, Xingxing Zhang, Yi Wu, and Furu Wei. 2021. Sequence level contrastive learning for text summarization.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter J. Liu. 2020. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. In *Proceedings of the 37th International Conference on Machine Learning*, ICML'20. JMLR.org.

Wen Zhang, Yang Feng, Fandong Meng, Di You, and Qun Liu. 2019. Bridging the gap between training and inference for neural machine translation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4334–4343, Florence, Italy. Association for Computational Linguistics.

Ming Zhong, Pengfei Liu, Yiran Chen, Danqing Wang, Xipeng Qiu, and Xuanjing Huang. 2020. Extractive summarization as text matching. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6197–6208, Online. Association for Computational Linguistics.

A Datasets

CNN/DailyMail (Hermann et al., 2015) is a widely used summarization benchmark with 93k and 220k articles from CNN and DailyMail newspapers, respectively. We use its non-anonymized version and follow the conventional preprocessing steps (See et al., 2017). We have 287,113 articles for training, 13,368 for validation, and 11,490 for testing.

XSum (Narayan et al., 2018) is a highly abstractive summarization benchmark consisting of articles from BBC and their one-sentence summary. We have 204,045 articles for training, 11,332 for validation, and 11,334 for testing.

B Implementation Details

The implementation of this paper is based on the Transformers library (Wolf et al., 2020).

Parameter	CNN/DailyMail	XSum
Training Parameters		
batch size	2048	1024
warm-up steps	500	500
training steps	50000	70000
learning rate	3e-5	3e-5
lr scheduler	Polynomial	Polynomial
optimizer	Adam	Adam
label smoothing	0.1	0.1
weight decay	0.01	0.01
max encoder input length	1024	512
max decoder input length	128	64
Sequence Generation Parameters		
beam size	3	5
max-len	128	64
min-len	0	0
length penalty	1.0	1.0

Table 3: Parameters for BART and PEGASUS on different summarization benchmarks. For PEGASUS, we change the learning rate to 1e-4 instead.

C Effects of α and β

Please refer to Figure 5 and Figure 6.

D Generation Examples

Please refer to Table 4.

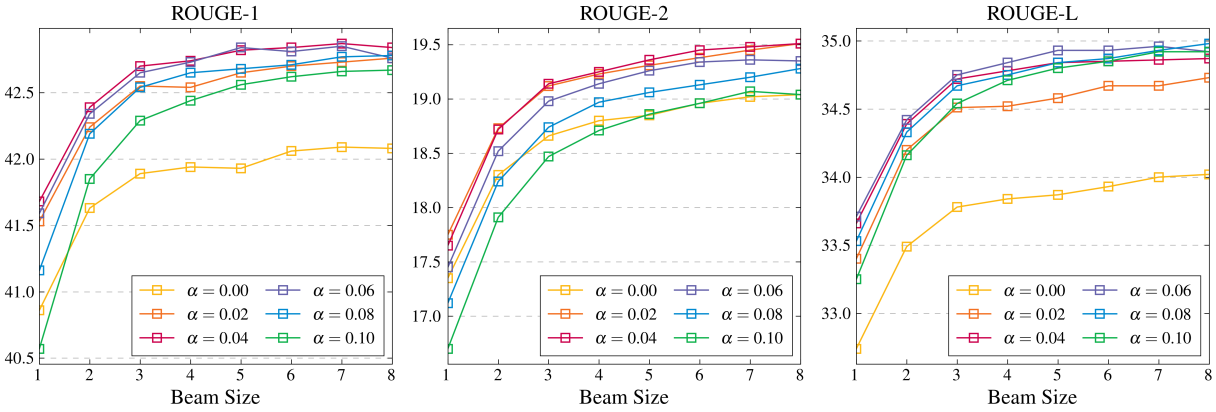

Figure 5: ROUGE scores of $\text{BART}_{\text{base}}$ with varying α and beam size on the XSum benchmark. The model with $\alpha = 0.04$ has a more balanced performance across different ROUGE f-measures compared with other settings. We can also observe a convergence of performance when the beam size is larger than 5.

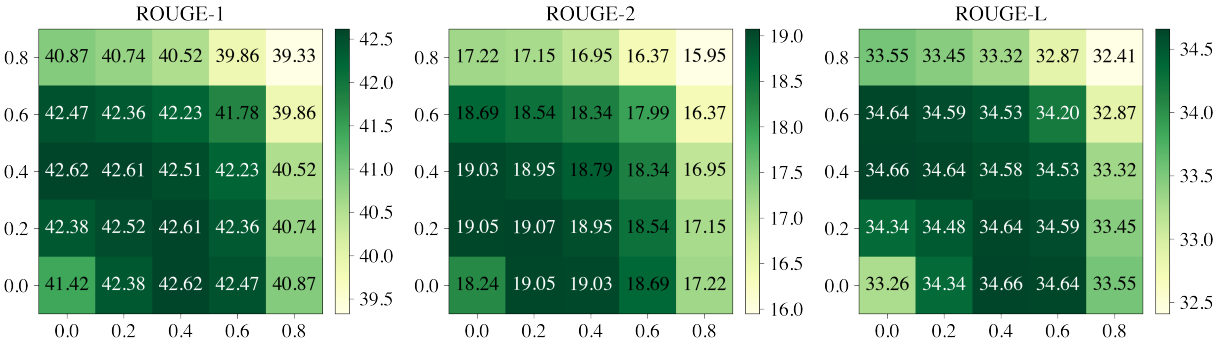

Figure 6: ROUGE scores of $\text{BART}_{\text{base}}$ on XSum when applying different β on two decoder input sequences during training. We set the augmentation rate to 1.0 on each side and follow the setting of augmenting with the mask token described in Eq. 2. The setting of (0.0, 0.4) results in the best performance across all ROUGE f-measures. Also, applying the same β on both sides of the decoder inputs underperforms the (0.0, 0.4) one, which coincides with Figure 2. The tables are symmetric and triangular since the combination of two β is commutative.

XSum	
Document	Pakistan's telecoms regulator said the ban was no longer necessary because Google, which owns YouTube, had now launched a Pakistan-specific version. YouTube has denied claims that the authorities can filter content. Many young Pakistanis have welcomed the lifting of the ban but some activists want details of the deal with Google. They say there should be greater transparency of the terms agreed between Google and the government. A Pakistan Telecommunication Authority (PTA) official confirmed to the BBC that all internet service providers had been directed to open access to YouTube. The Pakistan Telecommunication Company Ltd posted on its Facebook page on Monday: "Welcome Back YouTube". Pakistan's ministry of information technology said: "Google has provided an online web process through which requests for blocking access of offending material can be made by the PTA to Google directly. "Google/YouTube will accordingly restrict access to the said offending material for users within Pakistan." However, a YouTube spokeswoman said government requests for the removal of content would not automatically be granted. "We have clear community guidelines, and when videos violate those rules, we remove them," she said. "In addition, where we have launched YouTube locally and we are notified that a video is illegal in that country, we may restrict access to it after a thorough review." ... ...
Ground Truth	Pakistan has unblocked the video sharing site, YouTube, more than three years after it was banned for posting a video deemed insulting to Islam.
BART	YouTube has been reinstated in Pakistan after the authorities lifted a ban on the video-sharing site in the country.
BART + Pairwise	YouTube has been reinstated in Pakistan after the authorities lifted a ban on the video-sharing site.
BART + Augmentation	YouTube has been banned in Pakistan for the first time in more than two years after the authorities lifted a ban on the site.
BART + R-TeaFor	Pakistan has lifted its ban on YouTube in Pakistan, after a deal with Google.
Document	The reactor at Yongbyon has been the source of plutonium for North Korea's nuclear weapons programme. The White House said North Korea should "focus instead on fulfilling its international obligations". The reactor was shut down in 2007 as part of a disarmament-for-aid deal. But Pyongyang vowed to restart it in 2013, following its third nuclear test and amid high regional tensions. White House spokesman Josh Earnest said the international community would not accept North Korea as a nuclear state. "We will work with our partners in the context of the six-party talks to try to return North Korea to a posture of fulfilling those commitments that they have made," he said. "We will repeat our call that North Korea should refrain from the irresponsible provocations that aggravate regional tension and should focus instead on fulfilling its international obligations and commitments." Six-nation talks involving South Korea, the US, China, Japan and Russia aimed at ending the North's nuclear programme have been stalled since 2009. Experts believe that, when fully operational, the Yongbyon reactor can make one nuclear bomb's worth of plutonium per year. A US think-tank said this year that satellite images suggested work had started at the Yongbyon complex. ... ...
Ground Truth	The US has warned North Korea to refrain from "irresponsible provocation" after the communist state said its main nuclear facility had resumed normal operations.
BART	The White House has urged North Korea to " refrain from the irresponsible provocations" it has been accused of, after the North restarted a nuclear reactor.
BART + Pairwise	The White House has urged North Korea to refrain from "irresponsible provocations" after it announced it had restarted a nuclear reactor.
BART + Augmentation	The White House says North Korea has restarted a nuclear reactor in its Yongbyon complex, but will not carry out "irresponsible provocations".
BART + R-TeaFor	The White House has urged North Korea to stop "irresponsible provocations" after it said it had restarted a nuclear reactor.

Table 4: Generated summaries by the variations of $\text{BART}_{\text{base}}$ on the XSum benchmark.