Alleviating Exposure Bias via Multi-level Contrastive Learning and Deviation Simulation in Abstractive Summarization

Jiawen Xie[1], **Qi Su**[1], **Shaoting Zhang**[2], **Xiaofan Zhang**[1,2,†]
[1]Shanghai Jiao Tong University, China
[2]Shanghai AI Laboratory, China

Abstract

Most Transformer based abstractive summarization systems have a severe mismatch between training and inference, i.e., *exposure bias*. From diverse perspectives, we introduce a simple multi-level contrastive learning framework for abstractive summarization (SimMCS) and a tailored sparse decoder self-attention pattern (SDSA) to bridge the gap between training and inference to improve model performance. Compared with previous contrastive objectives focusing only on the relative order of probability mass assigned to non-gold summaries, SimMCS additionally takes their absolute positions into account, which guarantees that the relatively high-quality (positive) summaries among them can be properly assigned high probability mass, and further enhances the capability of discriminating summary quality beyond exploiting potential artifacts of specific metrics. SDSA simulates the possible inference scenarios of deviation in the training phase to get closer to the ideal paradigm. Our approaches outperform the previous state-of-the-art results on two summarization datasets while just adding fairly low overhead. Further empirical analysis shows our model preserves the advantages of prior contrastive methods and possesses strong few-shot learning ability. Our code is available at https://github.com/xjw-nlp/SimMCS.

1 Introduction

Automatic text summarization (El-Kassas et al., 2021) is the task of condensing a piece of text into a shorter version while preserving its most salient information and overall meaning. There are two main research directions for text summarization: extractive summarization and abstractive summarization (Nenkova et al., 2011). Extractive summarization involves selecting salient text spans from source documents, while abstractive summarization generates concise summaries in a sequence-to-sequence manner (Liu and Lapata, 2019; Raffel et al., 2020). Recently, large pre-trained neural models, typically based on encoder-decoder Transformer (Liu and Lapata, 2019; Bao et al., 2020), have shown promising performance in abstractive summarization. These models are generally optimized using maximum likelihood estimation (MLE) in *teacher forcing* form (Bengio et al., 2015; Lamb et al., 2016) to maximize the predictive probability of the reference output given its prior gold sub-sequence. However, during inference, the models produce output based on the generated sub-sequence, which may contain errors. This mismatch between training and inference can negatively impact model performance and is known as *exposure bias* (Bengio et al., 2015; Ranzato et al., 2015).

To alleviate the ubiquitous issue while maintaining reasonable performance, a variety of methods from various perspectives have been proposed. Among them, sentence-level training (Shao et al., 2018; Paulus et al., 2018; Stiennon et al., 2020) aims to train the model using sentence-level evaluation metrics (e.g., ROUGE). Another direction of attempts involves forging rational schemes to inject noise or perturb reference output during the decoding stage of training (Venkatraman et al., 2015; Ning et al., 2023). The model trained in the noisy environment can perceive more states of inference. Notably, recent contrastive learning methods in abstractive summarization combine the strengths of these approaches. Their contrastive objectives guide the model to distinguish between summaries with different metric scores at the sentence level.

While the strong contrastive method (Liu et al., 2022) achieves good results by correlating the relative order of probabilities of system-generated summaries with their evaluation metrics, considering these summaries have considerably high relevance to their gold summary, low probabilities of high-quality summaries can result in a low probability being assigned to their gold summary. Since the

†Corresponding author. xiaofan.zhang@sjtu.edu.cn

Findings of the Association for Computational Linguistics: ACL 2023, pages 9732–9747

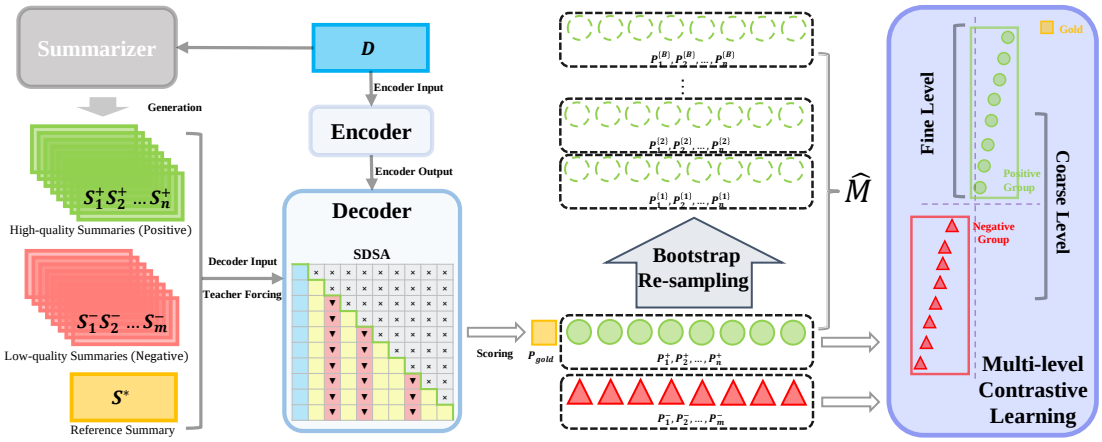

Figure 1: The overall training pipeline of SimMCS with SDSA. The positive group contains high-quality summaries generated by a strong summarizer given a document, while the negative group includes low-quality summaries weakly related to the document. We apply SDSA to the decoder self-attention module during training. We conduct B times of bootstrap re-sampling in the positive group and estimate $\hat{M}$ that represents the average adjacent spacing of positive samples. The ideal effect of our approach is exhibited in the right box.

probability of the reference summary is connected with the generative objective that preserves the generation ability of the model, the lack of constraint on the absolute position of probabilities of high-quality summaries could deteriorate model performance. To this end, we introduce a simple multi-level contrastive framework composed of fine and coarse contrastive components to address the *exposure bias* problem and explore further the potential power of large abstractive models in distinguishing semantic discrepancies of diverse summaries. Unlike existing margin ranking objectives (Zhong et al., 2020; Liu et al., 2021) which obtain a suitable margin with expensive grid search, to be more compatible with our whole design, we propose a bootstrap re-sampling process to acquire an adaptive margin in the fine contrastive stage. As for the coarse contrastive learning part, we design it mainly for two purposes - ensuring all probabilities of high-quality summaries at a properly high level and further discriminating between the semanticity of high- and low-quality summaries.

In addition, we introduce SDSA, a tailored sparse decoder self-attention pattern, to bridge the gap between training and inference. Unlike existing methods that perturb reference output in discrete space, SDSA operates in latent space. Specifically, we use this pattern during training while using standard attention during inference. Although the pattern may corrupt some salient token information of the reference, we argue that the resulting deviation could be acceptable since existing knowledge, including position information in the prior sub-sequence, is sufficient to predict the next token (Ethayarajh, 2019; Klafka and Ettinger, 2020).

Experiments on the CNN/DailyMail and XSum datasets show SimMCS consistently outperforms the prior state of the art. Furthermore, incorporating SimMCS with SDSA can further improve the model performance by a large margin. Further in-depth analysis indicates our system retains the advantages of previous contrastive methods and has strong few-shot learning ability.

2 Related Work

Transformer-based Pre-trained Model Recently, large Seq2Seq Transformers (Vaswani et al., 2017), which contain encoder self-attention, decoder self-attention, and decoder cross-attention, have achieved promising performance in the NLP domain, including text summarization (Song et al., 2019; Raffel et al., 2020). These models are pre-trained using a variety of self-supervised objectives and fine-tuned with structured losses in downstream tasks. For example, BART (Lewis et al., 2020), a denoising autoencoder, is pre-trained to reconstruct original text spans corrupted with an arbitrary noising function such as text infilling. PEGASUS (Zhang et al., 2020) is distinguished by its specifically tailored self-supervised pre-training objective for the summarization task. In PEGASUS, salient text spans are removed or masked from the original document, and the model aims to restore the remaining text spans to their original form. We

use these models as backbones in our work.

Mitigating Exposure Bias for Abstractive Summarization In the NLG domain, *exposure bias* is widespread and has received much attention from researchers (Daumé et al., 2009; Ross et al., 2011; Bengio et al., 2015; Wiseman and Rush, 2016; Zhang et al., 2019b; Ziegler et al., 2019). In abstractive summarization, Kryściński et al. (2018) introduces a reinforcement learning method with the ROUGE metric as a reward to encourage the generation of novel phrases. Inspired by Generative Adversarial Networks, Scialom et al. (2020) proposes a novel approach for sequence generation, in which the discriminator is integrated into a beam search (Tillmann and Ney, 2003; Li et al., 2016; Wiseman et al., 2017; Chen et al., 2018).

Contrastive Learning Contrastive learning has been widely confirmed to effectively boost model performance by allowing the model to distinguish between the quality of diverse samples (Chuang et al., 2020). Recently the method has shown promising performance in natural language generation tasks such as text summarization (Cao and Wang, 2021) and machine translation (Yang et al., 2019; Pan et al., 2021). In fact, these contrastive examples can be constructed using rule- or model-based methods, with the latter able to produce text examples closer to human-generated ones and forge more natural contrastive schemes. On the other hand, contrastive learning can be performed in latent or discrete space. For instance, Gao et al. (2021) introduces a contrastive learning framework into the representation of sentence embeddings and greatly advances state-of-the-art results. Liu et al. (2022) adopts the discriminative re-ranking over generated summaries in discrete space like other works (Shen et al., 2004; Och et al., 2004; Mizumoto and Matsumoto, 2016; Lee et al., 2021).

Bootstrap Re-sampling The bootstrap approach is a collection of sample reuse techniques designed to estimate sampling variances, confidence intervals, and other properties of statistics (Stine, 1989; Efron, 1992; Diciccio and Efron, 1992). Compared to traditional standard approaches, these techniques have fewer requirements and assumptions while achieving better performance and providing insight into many problems.

3 Methodology

Our complete design consists of a generative objective, multi-level contrastive learning framework SimMCS, and a sparse decoder self-attention pattern SDSA. During training, we train the abstractive model according to the training pipeline shown in Fig. 1. The reference summary is only used in the generative objective, while other types of summaries are used for contrastive learning. At the inference stage, the optimized model generates summaries in a conventional manner, using only the source document as input to its encoder.

3.1 Generative Objective

The training objective for summary generation consists of a sequence of token decisions made in an auto-regressive manner. This is formulated as a product of decision probabilities corresponding to specified tokens. Given a document $\mathcal{D} = (d_1, d_2, \cdots, d_{|\mathcal{D}|})$ and its summary $\mathcal{S} = (s_1, s_2, \cdots, s_{|\mathcal{S}|})$, we estimate the following conditional probability:

$$p_\theta(\mathcal{S}|\mathcal{D}) = \prod_{t=1}^{|\mathcal{S}|} p(s_t|s_{<t}, \mathcal{D}; \theta), \quad (1)$$

where $|\mathcal{S}|$ stands for the number of tokens in summary $\mathcal{S}$, θ represents the model parameters and $s_{<t}$ denotes all tokens prior to the position t.

In fact, most works based on Seq2Seq Transformers minimize the negative log-likelihood (NLL) of reference summaries. Following prior works, given current model parameters θ and a set of N document-reference pairs $\{\mathcal{D}^{(i)}, \mathcal{S}^{*(i)}\}_{i=1}^{N}$, our generative objective is as follows:

$$\mathcal{L}_{ref}(\theta) = -\frac{1}{N}\sum_{i=1}^{N} \frac{\log p_\theta\left(\mathcal{S}^{*(i)}|\mathcal{D}^{(i)}\right)}{|\mathcal{S}^{*(i)}|}. \quad (2)$$

Following previous works, during the practical fine-tuning, we transform the generative objective in Eq. 2 to a label smoothed cross-entropy loss (Szegedy et al., 2016; Pereyra et al., 2017) with the smoothing parameter set to 0.1.

3.2 Multi-level Contrastive Learning

Our multi-level contrastive learning framework, SimMCS, is designed for abstractive summarization and consists of fine and coarse contrastive components. Compared to recent contrastive methods that operate at a single level, SimMCS combines different contrastive signals in a natural way to further distinguish the semantic quality of summaries.

For each data point containing a source document, a reference summary, n system-generated

summaries (positive), and m randomly selected summaries weakly correlated with the reference (negative), namely, $\mathcal{D}, \mathcal{S}^*, (\mathcal{S}_1^+, \mathcal{S}_2^+, \cdots, \mathcal{S}_n^+), (\mathcal{S}_1^-, \mathcal{S}_2^-, \cdots, \mathcal{S}_m^-)$, we divide the n positive summaries and the m negative summaries into positive and negative groups respectively.

Similar to Eq. 1, we calculate the probability mass $P_{\mathcal{S}}$ corresponding to summary $\mathcal{S}$ as follows:

$$P_{\mathcal{S}} = \frac{\log p_\theta(\mathcal{S}|\mathcal{D})}{|\mathcal{S}|^\beta}, \tag{3}$$

where hyper-parameter β represents the degree of length penalty (Wu et al., 2016). Accordingly the probability mass $P_{\mathcal{S}}$ ranges from $-\infty$ to 0.

According to Eq. 3, we can obtain the probability mass of summaries in positive and negative groups to participate in the following contrastive objectives.

3.2.1 Fine Contrastive Learning

At the fine level, we consider the coordination of model-predicted probabilities and the quality of in-group summaries. Since measuring the quality of summaries using evaluation metrics such as ROUGE (Lin, 2004), BERTScore (Zhang et al., 2019a), and BARTScore (Yuan et al., 2021) involves non-trivial overhead, we only inject the fine contrastive signal from the positive group into our training procedure. While the lack of a fine contrastive signal from the negative group may weaken the ability of the model to rank low-quality summaries, we speculate that this trade-off is acceptable as the generation process during inference mainly requires comparison among relatively high-quality candidates for a strong summarizer.

To simplify the following expression, we sort the positive summaries in descending order by metric scores. That is, given a specific metric $\mathcal{M}$, $\mathcal{M}(\mathcal{S}^*, \mathcal{S}_i^+) > \mathcal{M}(\mathcal{S}^*, \mathcal{S}_j^+), \forall i, j, i < j$. The model-predicted probabilities $P_{\mathcal{S}_1^+}, P_{\mathcal{S}_2^+}, \cdots, P_{\mathcal{S}_n^+}$ correspond to the sorted summaries. To encourage the model to assign higher probabilities to summaries with higher metric scores, we formulate the following objective:

$$\mathcal{L}_{fine} = \sum_{i=1}^{n} \sum_{j=i+1}^{n} \max\{P_{\mathcal{S}_j^+} - P_{\mathcal{S}_i^+} + \lambda_{ij}, 0\}, \tag{4}$$

$$\lambda_{ij} = (j - i) \times \lambda, \tag{5}$$

where λ represents the unit margin. λ_{ij} represents the threshold judging whether the difference of $P_{\mathcal{S}_j^+}$ and $P_{\mathcal{S}_i^+}$ engages in backpropagation.

Algorithm 1: Margin Estimation

Input: Sample points $\{P_1, P_2, \cdots, P_n\}$
Input: Significance level α
Output: Estimated margin $\hat{M}$

$\hat{F}_n(x) = \frac{\sum_{i=1}^{n} \mathbb{I}(P_i \leq x)}{n}$;
$statistics = []$;
$M_n \leftarrow g(P_1, P_2, \cdots, P_n)$;
for $i \leftarrow 1$ **to** B **do**
  $P_{1,i}^*, P_{2,i}^*, \cdots, P_{n,i}^* \leftarrow \hat{F}_n$;
  $M_{n,i}^* \leftarrow g(P_{1,i}^*, P_{2,i}^*, \cdots, P_{n,i}^*)$;
  $statistics.append(M_{n,i}^*)$;
end
$\hat{F}_{boot}(x) = \frac{\sum_{i=1}^{B} \mathbb{I}(M_{n,i}^* \leq x)}{B}$;
$M_{n,\alpha/2}^* = \hat{F}_{boot}^{-1}(\alpha/2)$;
$M_{n,1-\alpha/2}^* = \hat{F}_{boot}^{-1}(1-\alpha/2)$;
$\hat{M} = 2M_n - \frac{M_{n,\alpha/2}^* + M_{n,1-\alpha/2}^*}{2}$;
return $\hat{M}$;

Margin Estimation with Bootstrap Re-sampling λ in Eq. 5 is the unit scale of the threshold. On account of our ultimate goal of comprehensively attending to both relative order and absolute position of probabilities, their valid range probably changes as the training progresses. Therefore, instead of framing λ as a hyper-parameter, we estimate it with bootstrap re-sampling to obtain a more representative margin.

Algorithm 1 shows the bootstrap re-sampling procedure. $P_{\mathcal{S}_1^+}, P_{\mathcal{S}_2^+}, \cdots, P_{\mathcal{S}_n^+}$ are regarded as sample points following certain population distribution F. $\hat{F}_n$ stands for empirical distribution consisting of these points. $g(\cdot)$ aims to compute statistics representing unit margin given the sample points properly. Accordingly, we perform bootstrap re-sampling to acquire B bootstrap statistics. Next, we calculate $1 - \alpha$ confidence interval for λ in terms of the bootstrap statistics. Generally, there are mainly three types of methods to estimate bootstrap confidential interval (Efron, 1979): normal interval, percentile interval, and pivotal interval. We choose the last to perform the estimation. Note the algorithm does not involve backpropagation. More details are shown in Appendix A.2.

3.2.2 Coarse Contrastive Learning

In addition to re-ranking the summaries in the positive group, we argue that it is also important to as-

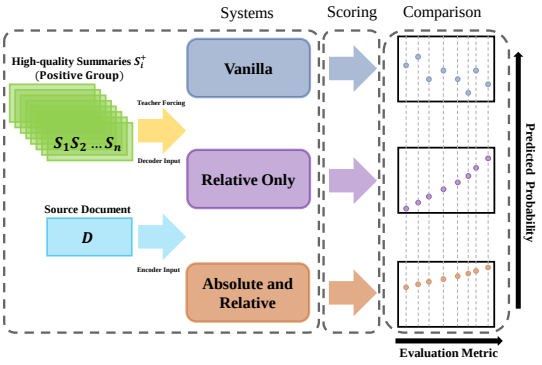

Figure 2: Comparison of reranking high-quality summaries. The blue is only trained with MLE loss; The purple represents the model that additionally considers the relative order of probabilities assigned to these summaries; The brown (ours) takes the relative order and absolute position of these probabilities into account.

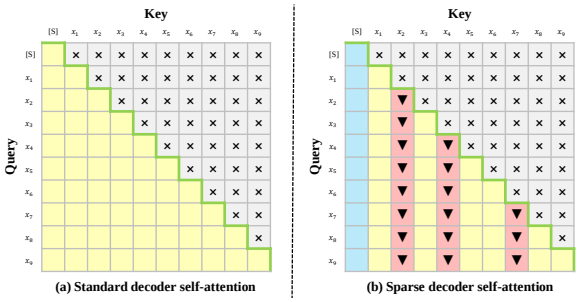

Figure 3: Comparison of two kinds of decoder self-attention patterns. Text span $[x_1, x_2, \cdots, x_9]$ is prepended with start token [S] as input and appended with end token [E] as output. Grey areas are masked out with the causal mask (ignoring the padding mask). Compared to (a), in (b) we mask out attention weights in red areas corresponding to randomly selected token positions on the "Key" side. Particularly, note the blue areas should not be involved since the start token exists permanently in inference time.

sign properly high probability mass to high-quality summaries. To this end, we introduce a new contrastive objective with a dual purpose: first, as a constraint to ensure that the probability mass of high-quality summaries is at a relatively high level; and second, as a signal enabling the system to further distinguish between the semantic discrepancy of high- and low-quality summaries.

$$P_{pos} = \frac{\sum_{i=1}^{n} w_i P_{\mathcal{S}_i^+}}{\sum_{i=1}^{n} w_i}, P_{neg} = \frac{1}{m} \sum_{j=1}^{m} P_{\mathcal{S}_j^-}, \quad (6)$$
$$\mathcal{L}_{coarse} = \log(1 + e^{P_{neg} - \xi P_{pos}}),$$

where P_{pos} denotes the weighted average probabilities of positive summaries, with higher quality summaries having greater weight w_i. P_{neg} is the average of probability mass assigned to negative summaries. ξ indicates the strength of constraint on P_{pos}.

Through the combination of fine and coarse contrastive signals, our contrastive loss is expected to keep the model-predicted probability mass of high-quality summaries at a properly high level to prevent degradation of model performance. Meanwhile, the aid of the fine contrastive signal allows the model to retain its ability to perceive the quality discrepancy of positive summaries (See Fig. 2).

To preserve both the generation and evaluation abilities of the model, we combine the multiple objectives above into a universal loss function (Edunov et al., 2018):

$$\mathcal{L}_{mul} = \mathcal{L}_{ref} + \gamma_1 \mathcal{L}_{fine} + \gamma_2 \mathcal{L}_{coarse}, \quad (7)$$

where γ_1 and γ_2 are the weights of fine contrastive loss and coarse contrastive loss respectively.

3.3 Sparse Decoder Self Attention

A Seq2Seq Transformer comprises three core attention modules: encoder self-attention, decoder self-attention, and decoder cross-attention. During training, the conventional decoder self-attention depicted in Fig. 3 (a) is crucial for the occurrence of the aforementioned mismatch (Arora et al., 2022).

To alleviate this ubiquitous issue in abstractive summarization, we have tailored a simple decoder self-attention pattern shown in Fig. 3 (b) to simulate the potential deviation that may arise during the inference phase. Specifically, during training, we mask out attention weights corresponding to arbitrarily selected token positions on the "Key" side (excluding the start token) while using the standard attention mechanism during inference. In this way, the system can learn to maximize the predictive probability of the reference output based on its previous suboptimal sub-sequence. It is important to note that the start token should not be masked since it is always present during inference.

Mathematically, we formulate a straightforward masking strategy that mimics the accumulation of errors during inference for the decoder self-attention pattern (Zhang et al., 2019b; Bonavita and Laloyaux, 2020). The mask ratio r for each token within a sequence is computed as follows:

$$r_i = i \times \frac{\mathbf{MR}}{len - 1},$$

where i represents the token position (starting from 0), *len* denotes the sequence length, and **MR** deter-

mines the degree of sparsity of our tailored decoder self-attention for training.

4 Experiments

Here we provide a brief overview of the datasets, baselines, implementation, and evaluation. More experimental details are provided in Appendix A.

4.1 Datasets

In our settings, we conduct the experiments on two single document summarization datasets, CNN/DailyMail (Hermann et al., 2015; Nallapati et al., 2016) and XSum (Narayan et al., 2018).

CNNDM[1] The CNN/DailyMail dataset (CNNDM) is a large-scale news dataset containing 93k and 220k news articles paired with associated highlights from the CNN and DailyMail websites, respectively. Following previous works (Nallapati et al., 2016; See et al., 2017; Liu and Lapata, 2019; Liu et al., 2022), we treat the highlights as summaries, and divide the article-summary pairs into a training set (287,227 samples), validation set (13,368 samples) and test set (11,490 samples).

XSum[2] The XSum dataset contains exceedingly abstractive summaries (i.e., single-sentence summaries) that are written professionally and collected from the British Broadcasting Corporation (BBC). We use standard splitting approaches to obtain 204,045 samples for training, 11,332 samples for validation, and 11,334 samples for testing.

4.2 Baselines

We intend to compare our experimental results with anterior-related works that exhibit outstanding performance. In particular, **BART** (Lewis et al., 2020) is a standard large pre-trained language model for sequence generation. Compared with BART, **PEGASUS** (Zhang et al., 2020) has a tailored pre-training objective for abstractive text summarization. **GSum** (Dou et al., 2021) is a general guided framework that could effectively take different kinds of external guidance as input. **SimCLS** (Liu and Liu, 2021) optimizes the text generation process with a two-stage method built on contrastive learning. **GOLD** (Pang and He, 2021) conducts generation by off-policy learning from demonstrations. **SeqCo** (Xu et al., 2021) regards the document, its reference summary and its candidate summaries as different views of the same mean representation and maximizes the similarities between them during training. **ConSum** (Sun and Li, 2021) remedies *exposure bias* problem through decreasing the likelihood of low-quality summaries while increasing the likelihood of reference summaries. **SummaReranker** (Ravaut et al., 2022) learns to select a high-quality summary from a collection of candidate summaries via applying re-ranking to a second-stage model. **BRIO** (Liu et al., 2022) introduces a new paradigm that assumes non-deterministic distributions instead of the deterministic distribution of gold summary.

Model	R-1	R-2	R-L	BS	BaS
BART	44.16	21.28	40.90	-	-
BART*	44.11	20.79	40.42	87.95	-3.91
PEGASUS	44.17	21.47	41.11	-	-
GSum	45.94	22.32	42.48	-	-
ConSum	44.53	21.54	41.57	-	-
SeqCo	45.02	21.80	41.75	-	-
GOLD-p	45.40	22.01	42.25	-	-
GOLD-s	44.82	22.09	41.81	-	-
SimCLS	46.67	22.15	43.54	66.14	-
SummaReranker	47.16	22.55	43.87	87.74	-
BRIO-Ctr	47.28	22.93	44.15	-	-
BRIO-Mul	47.78	23.55	44.57	-	-
SimMCS-Std	48.16	24.08	44.65	89.20	-3.58
SimMCS-SDSA	**48.38**	**24.17**	**44.79**	**89.31**	**-3.50**

Table 1: Average results on CNNDM test set. "*" is the result of our own evaluation script. R-1/2/L are the ROUGE-1/2/L F1 score ($p < 0.01$). BS and BaS refer to the neural model-based metrics BERTScore and BARTScore respectively. The best results are bolded.

4.3 Implementation Details

Our implementation is mainly based on *PyTorch* and *Transformers* library (Wolf et al., 2020), as well as 4 NVIDIA RTX 3090 GPUs.

Backbone Settings In accordance with prior works, we use BART_{Large}[3] with 12 layers each for the encoder and decoder, and PEGASUS_{Large}[4] with 16 encoder layers and 16 decoder layers as our backbones. In particular, the hidden size of each layer is 1024, which is converted into 16 attention heads with a hidden unit size of 64 for multi-head attention.

Training and Inference For each document, we obtain 16 summaries generated by a summarizer as positive summaries and 2 different human-generated summaries weakly correlated with the document as negative summaries. All PLMs are trained using the Adam optimizer (Kingma and Ba,

[1] https://cs.nyu.edu/ kcho/DMQA/
[2] https://github.com/EdinburghNLP/XSum
[3] facebook/bart-large-cnn
[4] google/pegasus-xsum

Model	R-1	R-2	R-L	BS	BaS
BART	45.14	22.27	37.25	-	-
PEGASUS	47.21	24.56	39.25	-	-
PEGASUS*	47.49	24.35	40.22	89.68	-3.89
GSum	45.40	21.89	36.67	-	-
ConSum	47.34	24.67	39.40	-	-
SeqCo	45.65	22.41	37.04	-	-
GOLD-p	45.75	22.26	37.30	-	-
GOLD-s	45.85	22.58	37.65	-	-
SimCLS	47.61	24.57	39.44	69.81	-
SummaReranker	48.12	24.95	40.00	92.14	-
BRIO-Ctr	48.13	25.13	39.84	-	-
BRIO-Mul	49.07	25.59	40.40	-	-
SimMCS-Std	49.39	25.73	40.49	90.23	-3.77
SimMCS-SDSA	**49.48**	**25.77**	**40.52**	**90.31**	**-3.73**

Table 2: Average results on XSum test set. "*" is the result of our own evaluation script. R-1/2/L are the ROUGE-1/2/L F1 score ($p < 0.01$). BS and BaS refer to the neural model-based metrics BERTScore and BARTScore respectively. The best results are bolded.[6]

2014) with $\beta_1 = 0.9, \beta_2 = 0.999$, along with a learning rate scheduling. Especially, at the training stage, we leverage our proposed SDSA pattern as an alternative to standard decoder self-attention in all decoder layers.

During inference, as common wisdom, summaries are generated with beam search in an auto-regressive manner (Wiseman and Rush, 2016) given source documents. In addition, note that we employ standard decoder self-attention instead of SDSA at this stage.

4.4 Evaluations

In practice, we measure the quality of generated summaries using the popular metric ROUGE. On the test set of CNNDM and XSum, we report full-length F1-based ROUGE-1, ROUGE-2, and ROUGE-L scores computed with the standard ROUGE-1.5.5.pl script[5]. Furthermore, We also use two popular model-based metrics BERTScore and BARTScore to demonstrate the superiority of our approaches more comprehensively.

5 Discussion

We evaluate two variants of our contrastive framework **SimMCS**: (1) **SimMCS-Std** uses standard attention modules, and (2) **SimMCS-SDSA** instead leverages our SDSA pattern as an alternative during training. Specifically, we select BART on CNNDM and PEGASUS on XSum as our base models respectively.

[5]with -c 95 -r 1000 -n 2 -a -m arguments

[6]In the origin paper of SummaReranker, the BS result of the base model (i.e., PEGASUS) was 92.01, compared to which its best model improved only slightly.

Model	R-1	R-2	R-L	BS	BaS
SimMCS-SDSA	48.38	24.17	44.79	89.31	-3.50
w/o SDSA	-0.22	-0.09	-0.14	-0.11	-0.08
w/o Coarse-Ctr	-0.35	-0.43	-0.36	-0.70	-0.19
w/o Fine-Ctr	-2.25	-1.87	-1.80	-1.51	-0.78
w/o Boot	-0.06	+0.08	-0.13	-0.12	+0.01

Table 3: Ablation study results. Performance changes compared with the full model are reported. Larger decreases in metrics are shaded with darker red and larger increases in metrics are shaded with darker green.

5.1 Comparison Results

Tab. 1 compares the results of the baseline, previous work from literature, and our proposed approaches on the CNNDM test set. We first note that even SimMCS-Std has outperformed the previous state-of-the-art model that is built on the extra single-level contrastive signal. This is evidence that our multi-level contrastive framework effectively discriminates the quality of diverse summaries and gains better efficacy. In detail, SimMCS-Std reports higher ROUGE scores (48.16/24.08/44.65 R-1/2/L) and model-based scores (89.20/-3.58 BS/BaS), indicating our improvement beyond ROUGE, even though the quality of positive summaries is measured with ROUGE-1. Moreover, with SDSA, we further improve SimMCS on all metrics and establish new state-of-the-art results on CNNDM with SimMCS-SDSA (48.38/24.17/44.79 R-1/2/L, 89.31/-3.50 BS/BaS). In this case, we employ our SDSA rather than standard decoder self-attention in all decoder layers. We attribute the superior performance of SDSA to its ability to alleviate overfitting and bridge the gap between training and inference.

To demonstrate the effectiveness of our method beyond the CNNDM dataset. We also conduct experiments on another news dataset, XSum. As shown in Tab. 2, notably, our method surpasses previous baselines and achieves new state-of-the-art results. The trend is similar to that of CNNDM and shows a strong generalization of our methods.

5.2 Analysis

We further analyze the properties of our state-of-the-art method and compare it with other strong baselines on the CNNDM dataset to gain more insights.

Ablation Studies We verify the contributions of

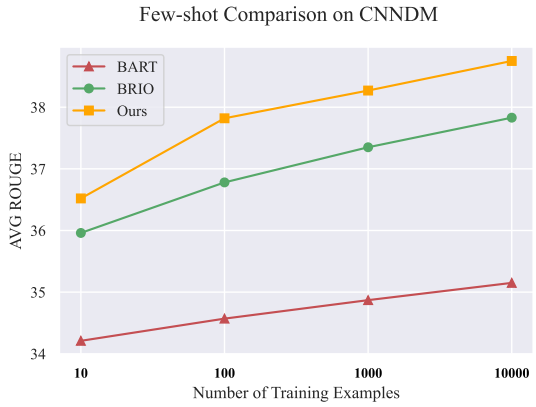

Figure 4: The AVG ROUGE scores (the average of R-1, R-2, and R-L F_1 scores) of various systems fine-tuned with 10, 100, 1000, and 10000 training examples.

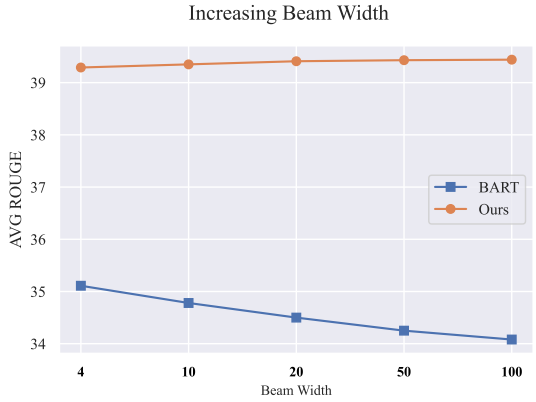

Figure 5: The AVG ROUGE scores (the average of R-1, R-2 and R-L F_1 scores) on CNNDM test set with different beam widths .

various components in SimMCS-SDSA on the CNNDM test set and show the ablation study results in Tab. 3. Specifically, we consider taking out SDSA, coarse contrastive learning (Coarse-Ctr), fine contrastive learning (Fine-Ctr), and margin estimation with bootstrap re-sampling (Boot), respectively. On the grounds of results, we can come to the conclusion that 1) removing SDSA, Coarse-Ctr, and Fine-Ctr substantially hurts performance, and 2) an adaptive margin under our SimMCS framework can improve model performance on most metrics. Considering that the proper margin in previous work is obtained with expensive grid search, our adaptive margin requires lower overhead for searching. The results in Fig. 4 present that our approach has a fairly strong few-shot learning capability.

Low-Resource Evaluation Considering that our approach can be applied to any Seq2Seq Transformer. Additionally, it is also imperative to possess sound few-shot learning ability since it seems to be unlikely to have a large amount of training data in actual application. To investigate how well our state-of-the-art model performs with limited training examples on downstream tasks, we explore test-set performance on CNNDM with varying numbers of training examples (e.g.,10, 100, 1000, and 10000). We compare our best system with BART and strong baseline BRIO.

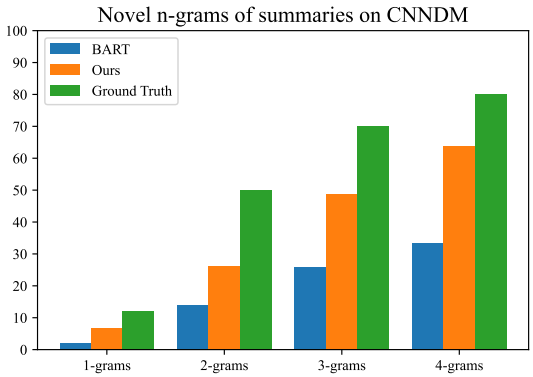

Figure 6: Novel n-grams on CNNDM dataset.

Increasing Beam Width Since our multi-level contrastive framework includes a fine contrastive signal to rerank the quality of summaries, it preserves the ability to coordinate candidate summaries and therefore improves the upper bound of performance with increasing beam width (i.e., the number of beams). We test our model performance with beam width 4, 10, 20, 50, and 100. As shown in Fig. 5, the larger the beam width, the better the performance on the CNNDM test set.

Abstractiveness We analyze the abstractiveness of generated summaries by calculating the percentage of novel n-grams, which are defined as those that appear in the summary but not in the associated source document (See et al., 2017; Liu and Lapata, 2019). As shown in Fig. 6, our state-of-the-art model generates more abstractive summaries than the base model BART in terms of all n-grams metrics. Additionally, Fig. 7 demonstrates that the summaries generated by our model effectively convey salient information and are closer to the reference summaries.

Case Study on CNNDM Fig. 7 displays several examples of summaries generated by SimMCS-SDSA and the base model BART, and their corresponding reference summaries. Specifically, SimMCS-SDSA is capable of identifying salient text spans that are overlooked by the base model. Furthermore, compared to the base summaries, the

System	Summary
Reference	brazilian neymar took to instagram to show off his skills to his followers . quick-footed barcelona attacker impressed for the filming camera. neymar will be hoping to show off that trickery away against sevilla. barcelona currently sit four points clear of rivals real madrid at the top.
BART	barcelona face sevilla in la liga on saturday. barcelona ace neymar posted a video on instagram showing off his skills. the former santos ace shows off his array of skills in just his shorts. the brazilian shows off a friend as he juggles the ball.
SimMCS-SDSA	barcelona travel to face sevilla in the la liga on saturday. neymar showed off his skills ahead of the trip to the ramon sanchez pizjuan. the former santos ace posted a video on instagram. barcelona are four points clear at the top of the table.
Reference	michelle filkins, 44, of west wareham has been charged with breaking and entering, larceny over $250, and the malicious destruction of property . she was arrested on april 17 after owner mark conklin found her sitting in his summer home. a neighbor told police he saw filkins outside with items from the house and that she appeared to be having a yard sale or giving the items away. police are asking anyone who received items from the home - including a lamp and a painting - to return them.
BART	michelle filkins, 44, of west wareham has been charged with breaking and entering, larceny over $250, and malicious destruction of property. she was discovered at the court street property in edgartown by owner mark conklin on april 17 after he found her sitting in his summer home. when he confronted her she claimed she owned the house. a construction worker in the neighborhood has told police that she appeared to be having a yard sale.
SimMCS-SDSA	michelle filkins, 44, of west wareham has been charged with breaking and entering, larceny over $250, and malicious destruction of property. she was discovered at the court street property in edgartown by owner mark conklin on april 17. a construction worker in the neighborhood said he saw her outside with items from the home. filkins was arrested on April 17 after owner found her sitting in his summer home.
Reference	1,500 people are attending the touching ceremony at cologne cathedral. among them are 500 relatives of those who died in germanwings crash . the doomed plane was 'deliberately' crashed by its 'depressed' co-pilot. cardinal woelki has urged compassion for all victims, including lubitz.
BART	memorial service held at cologne cathedral in west-german city this morning. cardinal rainer woelki urged forgiveness for victims including co-pilot andreas lubitz who is blamed for 'deliberately' crashing plane in french alps. 1,500 people attended the service including german chancellor angela merkel and german president joachim gauck. 150 candles were lit in memory of the victims at the historic cathedral.
SimMCS-SDSA	1,500 people attended memorial service at cologne cathedral in west-german city this morning. cardinal rainer woelki urged forgiveness for all of the victims - including co-pilot andreas lubitz blamed for 'deliberately' crashing the plane. 1,500 relatives of victims of germanwings air disaster in the french alps attended service. german president joachim gauck and german chancellor angela merkel among the ceremony. 150 candles were lit and flags flown at half-mast .

Figure 7: Examples of summaries generated by SimMCS-SDSA trained on CNNDM. The sentence in green is included in the SimMCS-SDSA summary, while the one in red is discarded.

summaries produced by our model exhibit fewer syntactical errors and are more closely aligned with the reference summaries.

6 Conclusion and Future Work

We introduce SimMCS, a simple multi-level contrastive learning framework for abstractive summarization that simultaneously considers the relative order and absolute position of probabilities assigned to high-quality summaries and further discriminates semantic discrepancy of summaries at different quality levels. Furthermore, we propose a simple yet empirically effective decoder self-attention pattern to alleviate *exposure bias* and improve model performance by a large margin. All our methods are not restricted to specific tasks or models and demonstrate strong generalization in conditional text generation tasks.

There are several directions to further exploit the potential power of our approaches. First, we believe that the margin estimation with bootstrap re-sampling could be more accurate and robust if given more probability mass. Second, it is also feasible to explore more sparse decoder self-attention mechanisms with diverse strategies. Finally, our methods could be extended to other text generation tasks such as machine translation.

Limitations

Since our contrastive framework requires the probability mass of various summaries given a source document, there is an extremely large consumption of GPU memory even if the batch size is small, which limits the scale of contrastive data and suppresses the potential of our method. Meanwhile, due to limited sample points (i.e., probability mass), our bootstrap re-sampling procedure is susceptible to outliers and cannot fully take advantage of this algorithm. In addition, like most abstractive summarization systems, our model does not attach importance to controllable text generation (Hu et al., 2017; Prabhumoye et al., 2020; He et al., 2020), which means that the generated text might contain redundant and incorrect information.

Ethical Considerations

While there is limited risk associated with our work, similar to existing abstractive summarization systems, there is no guarantee that the generated summaries are factually consistent and free from hallucination (Maynez et al., 2020; Kang and Hashimoto, 2020). Therefore caution is imperative when our system is applied to practical projects.

References

Kushal Arora, Layla El Asri, Hareesh Bahuleyan, and Jackie Cheung. 2022. Why exposure bias matters: An imitation learning perspective of error accumulation in language generation. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 700–710, Dublin, Ireland. Association for Computational Linguistics.

Hangbo Bao, Li Dong, Furu Wei, Wenhui Wang, Nan Yang, Xiaodong Liu, Yu Wang, Jianfeng Gao, Song-

hao Piao, Ming Zhou, et al. 2020. Unilmv2: Pseudo-masked language models for unified language model pre-training. In *International Conference on Machine Learning*, pages 642–652. PMLR.

Samy Bengio, Oriol Vinyals, Navdeep Jaitly, and Noam Shazeer. 2015. Scheduled sampling for sequence prediction with recurrent neural networks. *Advances in neural information processing systems*, 28.

Massimo Bonavita and Patrick Laloyaux. 2020. Machine learning for model error inference and correction. *Journal of Advances in Modeling Earth Systems*, 12(12):e2020MS002232.

Shuyang Cao and Lu Wang. 2021. CLIFF: Contrastive learning for improving faithfulness and factuality in abstractive summarization. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6633–6649, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Yining Chen, Sorcha Gilroy, Andreas Maletti, Jonathan May, and Kevin Knight. 2018. Recurrent neural networks as weighted language recognizers. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2261–2271, New Orleans, Louisiana. Association for Computational Linguistics.

Ching-Yao Chuang, Joshua Robinson, Yen-Chen Lin, Antonio Torralba, and Stefanie Jegelka. 2020. Debiased contrastive learning. In *Advances in Neural Information Processing Systems*, volume 33, pages 8765–8775. Curran Associates, Inc.

Hal Daumé, John Langford, and Daniel Marcu. 2009. Search-based structured prediction. *Machine learning*, 75(3):297–325.

Thomas Diciccio and Bradley Efron. 1992. More accurate confidence intervals in exponential families. *Biometrika*, 79(2):231–245.

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.

Sergey Edunov, Myle Ott, Michael Auli, David Grangier, and Marc'Aurelio Ranzato. 2018. Classical structured prediction losses for sequence to sequence learning. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 355–364, New Orleans, Louisiana. Association for Computational Linguistics.

B. Efron. 1979. Bootstrap Methods: Another Look at the Jackknife. *The Annals of Statistics*, 7(1):1 – 26.

Bradley Efron. 1992. Bootstrap methods: another look at the jackknife. In *Breakthroughs in statistics*, pages 569–593. Springer.

Wafaa S El-Kassas, Cherif R Salama, Ahmed A Rafea, and Hoda K Mohamed. 2021. Automatic text summarization: A comprehensive survey. *Expert Systems with Applications*, 165:113679.

Kawin Ethayarajh. 2019. How contextual are contextualized word representations? Comparing the geometry of BERT, ELMo, and GPT-2 embeddings. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 55–65, Hong Kong, China. Association for Computational Linguistics.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. Simcse: Simple contrastive learning of sentence embeddings. *arXiv preprint arXiv:2104.08821*.

Junxian He, Wojciech Kryściński, Bryan McCann, Nazneen Rajani, and Caiming Xiong. 2020. Ctrlsum: Towards generic controllable text summarization. *arXiv preprint arXiv:2012.04281*.

Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. *Advances in neural information processing systems*, 28.

Zhiting Hu, Zichao Yang, Xiaodan Liang, Ruslan Salakhutdinov, and Eric P Xing. 2017. Toward controlled generation of text. In *International conference on machine learning*, pages 1587–1596. PMLR.

Daniel Kang and Tatsunori B. Hashimoto. 2020. Improved natural language generation via loss truncation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 718–731, Online. Association for Computational Linguistics.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Josef Klafka and Allyson Ettinger. 2020. Spying on your neighbors: Fine-grained probing of contextual embeddings for information about surrounding words. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4801–4811, Online. Association for Computational Linguistics.

Wojciech Kryściński, Romain Paulus, Caiming Xiong, and Richard Socher. 2018. Improving abstraction in text summarization. *arXiv preprint arXiv:1808.07913*.

Alex M Lamb, Anirudh Goyal ALIAS PARTH GOYAL, Ying Zhang, Saizheng Zhang, Aaron C Courville, and Yoshua Bengio. 2016. Professor forcing: A new algorithm for training recurrent networks. In *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc.

Ann Lee, Michael Auli, and Marc'Aurelio Ranzato. 2021. Discriminative reranking for 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 7250–7264, Online. Association for Computational Linguistics.

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.

Jiwei Li, Will Monroe, Alan Ritter, Dan Jurafsky, Michel Galley, and Jianfeng Gao. 2016. Deep reinforcement learning for dialogue generation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1192–1202, Austin, Texas. Association for Computational Linguistics.

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.

Yang Liu and Mirella Lapata. 2019. Text summarization with pretrained encoders. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3730–3740, Hong Kong, China. Association for Computational Linguistics.

Yixin Liu, Zi-Yi Dou, and Pengfei Liu. 2021. RefSum: Refactoring neural summarization. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1437–1448, Online. 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.

Yixin Liu, Pengfei Liu, Dragomir Radev, and Graham Neubig. 2022. Brio: Bringing order to abstractive summarization. *arXiv preprint arXiv:2203.16804*.

Joshua Maynez, Shashi Narayan, Bernd Bohnet, and Ryan McDonald. 2020. On faithfulness and factuality in abstractive summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1906–1919, Online. Association for Computational Linguistics.

Tomoya Mizumoto and Yuji Matsumoto. 2016. Discriminative reranking for grammatical error correction with statistical machine translation. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1133–1138, San Diego, California. Association for Computational Linguistics.

Ramesh Nallapati, Bowen Zhou, Cicero dos Santos, Çağlar Gulçehre, 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*, pages 280–290, Berlin, Germany. Association for Computational Linguistics.

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.

Ani Nenkova, Kathleen McKeown, et al. 2011. Automatic summarization. *Foundations and Trends® in Information Retrieval*, 5(2–3):103–233.

Mang Ning, Enver Sangineto, Angelo Porrello, Simone Calderara, and Rita Cucchiara. 2023. Input perturbation reduces exposure bias in diffusion models.

Franz Josef Och, Daniel Gildea, Sanjeev Khudanpur, Anoop Sarkar, Kenji Yamada, Alex Fraser, Shankar Kumar, Libin Shen, David Smith, Katherine Eng, Viren Jain, Zhen Jin, and Dragomir Radev. 2004. A smorgasbord of features for statistical machine translation. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics: HLT-NAACL 2004*, pages 161–168, Boston, Massachusetts, USA. 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*.

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, Łukasz Kaiser, and Geoffrey Hinton. 2017. Regularizing neural networks by penalizing confident output distributions. *arXiv preprint arXiv:1701.06548*.

Shrimai Prabhumoye, Alan W Black, and Ruslan Salakhutdinov. 2020. Exploring controllable text generation techniques. *arXiv preprint arXiv:2005.01822*.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, Peter J Liu, et al. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21(140):1–67.

Marc'Aurelio Ranzato, Sumit Chopra, Michael Auli, and Wojciech Zaremba. 2015. Sequence level training with recurrent neural networks. *arXiv preprint arXiv:1511.06732*.

Mathieu Ravaut, Shafiq Joty, and Nancy F Chen. 2022. Summareranker: A multi-task mixture-of-experts reranking framework for abstractive summarization. *arXiv preprint arXiv:2203.06569*.

Stephane Ross, Geoffrey Gordon, and Drew Bagnell. 2011. A reduction of imitation learning and structured prediction to no-regret online learning. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, volume 15 of *Proceedings of Machine Learning Research*, pages 627–635, Fort Lauderdale, FL, USA. PMLR.

Thomas Scialom, Paul-Alexis Dray, Sylvain Lamprier, Benjamin Piwowarski, and Jacopo Staiano. 2020. Discriminative adversarial search for abstractive summarization. In *International Conference on Machine Learning*, pages 8555–8564. PMLR.

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.

Chenze Shao, Yang Feng, and Xilin Chen. 2018. Greedy search with probabilistic n-gram matching for neural machine translation. *arXiv preprint arXiv:1809.03132*.

Libin Shen, Anoop Sarkar, and Franz Josef Och. 2004. Discriminative reranking for machine translation. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics: HLT-NAACL 2004*, pages 177–184, Boston, Massachusetts, USA. Association for Computational Linguistics.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2019. Mass: Masked sequence to sequence pre-training for language generation. *arXiv preprint arXiv:1905.02450*.

Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F Christiano. 2020. Learning to summarize with human feedback. *Advances in Neural Information Processing Systems*, 33:3008–3021.

Robert Stine. 1989. An introduction to bootstrap methods: Examples and ideas. *Sociological Methods & Research*, 18(2-3):243–291.

Shichao Sun and Wenjie Li. 2021. Alleviating exposure bias via contrastive learning for abstractive text summarization. *CoRR*, abs/2108.11846.

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. 2016. Rethinking the inception architecture for computer vision. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2818–2826.

Christoph Tillmann and Hermann Ney. 2003. Word Reordering and a Dynamic Programming Beam Search Algorithm for Statistical Machine Translation. *Computational Linguistics*, 29(1):97–133.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Arun Venkatraman, Martial Hebert, and J. Andrew Bagnell. 2015. Improving multi-step prediction of learned time series models. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, AAAI'15, page 3024–3030. AAAI Press.

Ashwin Vijayakumar, Michael Cogswell, Ramprasaath Selvaraju, Qing Sun, Stefan Lee, David Crandall, and Dhruv Batra. 2018. Diverse beam search for improved description of complex scenes. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

Sam Wiseman and Alexander M Rush. 2016. Sequence-to-sequence learning as beam-search optimization. *arXiv preprint arXiv:1606.02960*.

Sam Wiseman, Stuart Shieber, and Alexander Rush. 2017. Challenges in data-to-document generation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2253–2263, Copenhagen, Denmark. 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.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*.

Shusheng Xu, Xingxing Zhang, Yi Wu, and Furu Wei. 2021. Sequence level contrastive learning for text summarization. *CoRR*, abs/2109.03481.

Zonghan Yang, Yong Cheng, Yang Liu, and Maosong Sun. 2019. Reducing word omission errors in neural machine translation: A contrastive learning approach. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6191–6196, Florence, Italy. Association for Computational Linguistics.

Weizhe Yuan, Graham Neubig, and Pengfei Liu. 2021. Bartscore: Evaluating generated text as text generation. *Advances in Neural Information Processing Systems*, 34:27263–27277.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter Liu. 2020. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. In *International Conference on Machine Learning*, pages 11328–11339. PMLR.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019a. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.

Wen Zhang, Yang Feng, Fandong Meng, Di You, and Qun Liu. 2019b. Bridging the gap between training and inference for neural machine translation. *arXiv preprint arXiv:1906.02448*.

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.

Daniel M Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B Brown, Alec Radford, Dario Amodei, Paul Christiano, and Geoffrey Irving. 2019. Fine-tuning language models from human preferences. *arXiv preprint arXiv:1909.08593*.

A Appendix

A.1 Dataset Statistics

Dataset	Type			Avg. Words	
	Train	Valid	Test	Doc.	Sum.
CNNDM	287K	13K	11K	791.6	55.6
XSum	203K	11K	11K	429.2	23.3

Table 4: Statistics of used datasets.

A.2 Settings

In this paper, we follow recent works (Liu and Liu, 2021; Liu et al., 2022) and generate 16 candidate summaries as the positive group for each data sample using diverse beam search (Vijayakumar et al., 2018). We also randomly select 2 summaries that have a low correlation with the gold summary. On CNNDM, we use the pre-trained BART$_{Large}$[7] to conduct candidate summary generation, while for XSum we produce candidate summaries via PEGASUS$_{Large}$[8]. The generated candidate summaries are ordered based on their ROUGE-1 score. For model training, We employ the Adam optimizer with a dynamic learning rate:

$$lr = 2 \times 10^{-3} \min(step^{-0.5}, step \times warmup^{-1.5}),$$

where *warmup* indicates the warmup steps, which is set to 10000, *step* is the number of updating steps, and lr is the learning rate.

The length penalty factor β in Eq. 3 is assigned the same value as that used in the original beam search. With regards to the dynamic margin estimation with bootstrap re-sampling, as described in Algorithm 1, the function $g(\cdot)$ is defined as follows:

$$g(\cdot) = \frac{\max(\cdot) - \min(\cdot)}{\text{num}(\cdot) - 1},$$

where $\text{num}(\cdot)$ represents the number of sample points, $\max(\cdot)$ and $\min(\cdot)$ are the maximum and minimum values respectively in this sample group.

Since there are limited sample points (i.e., the probability mass of candidate summaries) we additionally provide a boundary constraint to mitigate the impact of outliers (See Tab. 5).

[7]The checkpoint is "facebook/bart-large-cnn" containing around 406M parameters.

[8]The checkpoint is "google/pegasus-xsum" containing around 568M parameters.

Dataset	β	ξ	γ_1	γ_2	BC	MR
CNNDM	2.0	2	100	0.1	$[1.0\times10^{-4}, 2.0\times10^{-3}]$	0.30
XSum	0.6	2	100	0.1	$[1.0\times10^{-4}, 0.1]$	0.15

Table 5: Hyper-parameter settings. **BC** indicates the boundary constraint on margin estimation; **MR** determines the degree of sparsity of our proposed sparse decoder self-attention pattern.

We also conduct an extensive search for the optimal values of γ_1 in the fine contrastive objective and γ_2 in the coarse contrastive objective respectively. Additionally, we investigate the constraint strength ξ and the mask ratio of decoder self-attention. All the hyper-parameter settings are reported in Tab. 5.

A.3 Evaluations

Prior to evaluation, all gold summaries and system outputs are converted to lowercase and tokenized using the PTB tokenizer[9]. For ROUGE metrics, we employ the standard ROUGE (Lin, 2004) Perl package[10]. With regards to BERTScore, we utilize the publicly available *bert-score* package[11] provided by the authors. Notably, since the results of BARTScore are highly sensitive to the selected scoring models, we consistently use the officially provided *bart-score*[12] for evaluation.

[9]PTB tokenizer.

[10]ROUGE-RELEASE-1.5.5

[11]https://github.com/Tiiiger/bert$_{s}core$

[12]https://github.com/neulab/BARTScore/tree/main/SUM

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
7 Limitations

☑ A2. Did you discuss any potential risks of your work?
8 Ethical Considerations

☑ A3. Do the abstract and introduction summarize the paper's main claims?
1

☒ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☑ Did you use or create scientific artifacts?

4

☑ B1. Did you cite the creators of artifacts you used?
4

☒ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
we use two public datasets

☑ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
4

☒ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
all the used datasets are downloaded from the official repository.

☑ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
4

☑ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
Appendix

C ☑ Did you run computational experiments?

4

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
Appendix

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
Appendix

☑ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
5

☑ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
4

D ☒ Did you use human annotators (e.g., crowdworkers) or research with human participants?

Left blank.

☐ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
No response.

☐ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
No response.

☐ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
No response.

☐ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
No response.

☐ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
No response.