Target-Side Augmentation for Document-Level Machine Translation

Guangsheng Bao[1,2], **Zhiyang Teng**[3], **and Yue Zhang**[*,2,4],
[1] Zhejiang University
[2] School of Engineering, Westlake University
[3] Nanyang Technological University
[4] Institute of Advanced Technology, Westlake Institute for Advanced Study
[2] {baoguangsheng, zhangyue}@westlake.edu.cn
[3] zhiyang.teng@ntu.edu.sg

Abstract

Document-level machine translation faces the challenge of data sparsity due to its long input length and a small amount of training data, increasing the risk of learning spurious patterns. To address this challenge, we propose a target-side augmentation method, introducing a data augmentation (DA) model to generate many potential translations for each source document. Learning on these wider range translations, an MT model can learn a smoothed distribution, thereby reducing the risk of data sparsity. We demonstrate that the DA model, which estimates the posterior distribution, largely improves the MT performance, outperforming the previous best system by 2.30 s-BLEU on News and achieving new state-of-the-art on News and Europarl benchmarks. Our code is available at https://github.com/baoguangsheng/target-side-augmentation.

1 Introduction

Document-level machine translation (Gong et al., 2011; Hardmeier et al., 2013; Werlen et al., 2018; Maruf et al., 2019; Bao et al., 2021; Feng et al., 2022) has received increasing research attention. It addresses the limitations of sentence-level MT by considering cross-sentence co-references and discourse information, and therefore can be more useful in the practical setting. Document-level MT presents several unique technical challenges, including significantly longer inputs (Bao et al., 2021) and relatively smaller training data compared to sentence-level MT (Junczys-Dowmunt, 2019; Liu et al., 2020; Sun et al., 2022). The combination of these challenges leads to increased data sparsity (Gao et al., 2014; Koehn and Knowles, 2017; Liu et al., 2020), which raises the risk of learning spurious patterns in the training data (Belkin et al., 2019; Savoldi et al., 2021) and hinders generalization (Li et al., 2021; Dankers et al., 2022).

To address these issues, we propose a target-side data augmentation method that aims to reduce sparsity by automatically smoothing the training distribution. The main idea is to train the document MT model with many plausible potential translations, rather than forcing it to fit a single human translation for each source document. This allows the model to learn more robust and generalizable patterns, rather than being overly reliant on features of particular training samples. Specifically, we introduce a data augmentation (DA) model to generate possible translations to guide MT model training. As shown in Figure 1, the DA model is trained to understand the relationship between the source and possible translations based on one observed translation (Step 1), and then used to sample a set of potentially plausible translations (Step 2). These translations are fed to the MT model for training, smoothing the distribution of target translations (Step 3).

We use standard document-level MT models including Transformer (Vaswani et al., 2017) and G-Transformer (Bao et al., 2021) for both our DA and MT models. For the DA model, in order to effectively capture a *posterior* target distribution given a reference target, we concatenate each source sentence with a latent token sequence as the new input, where the latent tokens are sampled from the observed translation. A challenge to the DA model is that having the reference translation in the input can potentially decrease diversity. To address this issue, we introduce the intermediate latent variable on the encoder side by using rules to generate n-gram samples, so that posterior sampling (Wang and Park, 2020) can be leveraged to yield diverse translations.

Results on three document-level MT benchmarks demonstrate that our method significantly outperforms Transformer and G-Transformer baselines, achieving an improvement of 1.33 and 1.75 s-BLEU on average, respectively, and the state-

[*] Corresponding author.

Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics
Volume 1: Long Papers, pages 10725–10742
July 9-14, 2023 ©2023 Association for Computational Linguistics

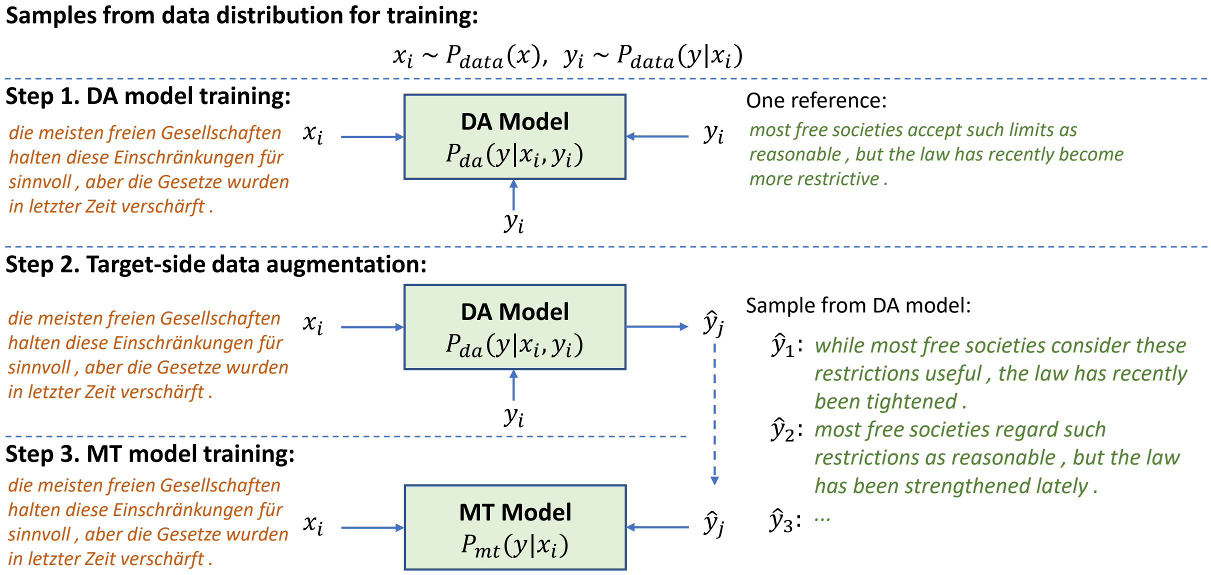

Samples from data distribution for training:
$x_i \sim P_{data}(x),\ y_i \sim P_{data}(y|x_i)$
Step 1. DA model training:
die meisten freien Gesellschaften halten diese Einschränkungen für sinnvoll , aber die Gesetze wurden in letzter Zeit verschärft .
x_i
DA Model
$P_{da}(y|x_i, y_i)$
y_i
y_i
One reference:
most free societies accept such limits as reasonable , but the law has recently become more restrictive .
Step 2. Target-side data augmentation:
die meisten freien Gesellschaften halten diese Einschränkungen für sinnvoll , aber die Gesetze wurden in letzter Zeit verschärft .
x_i
DA Model
$P_{da}(y|x_i, y_i)$
y_i
$\hat{y}_j$
Sample from DA model:
$\hat{y}_1$: while most free societies consider these restrictions useful , the law has recently been tightened .
$\hat{y}_2$: most free societies regard such restrictions as reasonable , but the law has been strengthened lately .
$\hat{y}_3$: ...
Step 3. MT model training:
die meisten freien Gesellschaften halten diese Einschränkungen für sinnvoll , aber die Gesetze wurden in letzter Zeit verschärft .
x_i
MT Model
$P_{mt}(y|x_i)$
$\hat{y}_j$

Figure 1: Illustration of target-side data augmentation (DA) using a very simple example. A DA model is trained to estimate the distribution of possible translations y given a source x_i and an observed target y_i, and the MT model is trained on the sampled translations $\hat{y}_j$ from the DA model for each source x_i. Effectively training the DA model with the target y_i, which is also a conditional input, can be challenging, but it is achievable after introducing an intermediate latent variable between the translation y and the condition y_i.

of-the-art results on News and Europarl. Further analysis shows that high diversity among generated translations and their low deviation from the gold translation are the keys to improved performance. To our knowledge, we are the first to do *target-side* augmentation to enrich *output* variety for document-level machine translation.

2 Related Work

Data augmentation (DA) increases training data by synthesizing new data (Van Dyk and Meng, 2001; Shorten and Khoshgoftaar, 2019; Shorten et al., 2021; Li et al., 2022). In neural machine translation (NMT), the most commonly used data augmentation techniques are **source-side augmentations**, including easy data augmentation (EDA) (Wei and Zou, 2019), subword regularization (Kudo, 2018), and back-translation (Sennrich et al., 2016a), which generates pseudo sources for monolingual targets enabling the usage of widely available monolingual data. These methods generate more source-target pairs with different silver source sentences for the same gold-target translation. On the contrary, **target-side augmentation** is more challenging, as approaches like EDA are not effective for the target side because they corrupt the target sequence, degrading the autoregressive modeling of the target language.

Previous approaches on target-side data augmentation in NMT fall into three categories. The first is based on *self-training* (Bogoychev and Sennrich, 2019; He et al., 2019; Zoph et al., 2020), which generates pseudo translations for monolingual source text using a trained model. The second category uses either a pre-trained language model (Fadaee et al., 2017; Wu et al., 2019) or a pre-trained generative model (Raffel et al., 2020; Khayrallah et al., 2020) to generate *synonyms* for words or *paraphrases* of the target text. The third category relies on reinforcement learning (Norouzi et al., 2016; Wang et al., 2018), introducing a reward function to evaluate the quality of translation candidates and to regularize the likelihood objective. In order to explore possible candidates, a sampling from the model distribution or random noise is used. Unlike these approaches, our method is a target-side data augmentation technique that is trained using supervised learning and does not rely on external data or large-scale pretraining. More importantly, we generate document-level instead of word, phrase, or sentence-level alternatives.

Previous target-side input augmentation (Xie et al., 2022) appears to be similar to our target-side augmentation. However, besides the literal similarity, they are quite different. Consider the token prediction $P(y_i|x, y_{<i})$. The target-side input augmentation augments the condition $y_{<i}$ to increase the model's robustness to the conditions,

which is more like source-side augmentation on condition x. In comparison, target-side augmentation augments the target y_i, providing the model with completely new training targets.

Paraphrase models. Our approach generates various translations for each source text, each of which can be viewed as a paraphrase of the target. Unlike previous methods that leverage paraphrase models for improving MT (Madnani et al., 2007; Hu et al., 2019; Khayrallah et al., 2020), our DA model exploits parallel corpus and does not depend on external paraphrase data, similar to Thompson and Post (2020). Instead, it takes into account the source text when modeling the target distribution. More importantly, while most paraphrase models operate at the sentence level, our DA model can generate translations at the document level.

Conditional auto-encoder. The DA model can also be seen as a conditional denoising auto-encoder (c-DAE), where the latent variable is a noised version of the ground-truth target, and the model is trained to reconstruct the ground-truth target from a noisy latent sequence. c-DAE is similar to the conditional variational autoencoder (c-VAE) (Zhang et al., 2016; Pagnoni et al., 2018), which learns a latent variable and generates diverse translations by sampling from it. However, there are two key differences between c-VAE and our DA model. First, c-VAE learns both the prior and posterior distributions of the latent variable, while the DA model directly uses predefined rules to generate the latent variable. Second, c-VAE models the prior distribution of the target, while the DA model estimates the posterior distribution.

Sequence-level knowledge distillation. Our DA-MT process is also remotely similar in form to sequence-level knowledge distillation (SKD) (Ba and Caruana, 2014; Hinton et al.; Gou et al., 2021; Kim and Rush, 2016; Gordon and Duh, 2019; Lin et al., 2020), which learns the data distribution using a large teacher and distills the knowledge into a small student by training the student using sequences generated by the teacher. However, our method differs from SKD in three aspects. First, SKD aims to compress knowledge from a large teacher to a small student, while we use the same or smaller size model as the DA model, where the knowledge source is the training data rather than the big teacher. Second, the teacher in SKD estimates the prior distribution of the target given source, while our DA model estimates the posterior distribution of the target given source and an observed target. Third, SKD generates one sequence for each source, while we generate multiple diverse translations with controlled latent variables.

3 Target-Side Augmentation

The overall framework is shown in Figure 1. Formally, denote a set of training data as $D = \{(x_i, y_i)\}_{i=1}^{N}$, where (x_i, y_i) is the i-th source-target pair and N is the number of pairs. We train a data augmentation (DA) model (Section 3.1) to generate samples with new target translations (Section 3.2), which are used to train an MT model (Section 3.3).

3.1 The Data Augmentation Model

We learn the posterior distribution $P_{da}(y|x_i, y_i)$ from parallel corpus by introducing latent variables

$$P_{da}(y|x_i, y_i) = \sum_{z \in \mathcal{Z}_i} P_{\varphi}(y|x_i, z) P_{\alpha}(z|y_i), \quad (1)$$

where z is the latent variable to control the translation output and $\mathcal{Z}_i$ denotes the possible space of z, φ denotes the parameters of the DA model, and α denotes the hyper-parameters for determining the distribution of z given y_i.

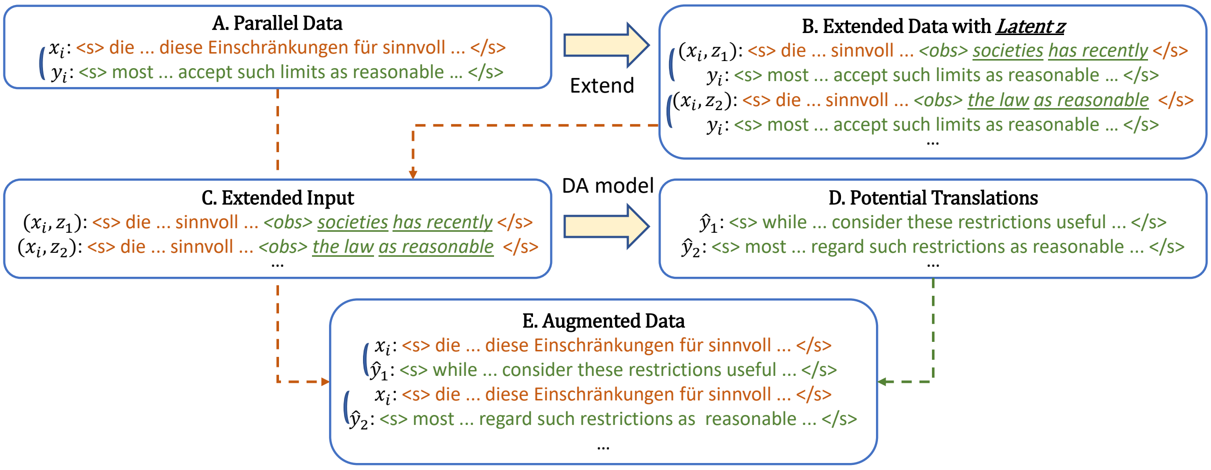
The space $\mathcal{Z}_i$ of possible z is exponentially large compared to the number of tokens of the target, making it intractable to sum over $\mathcal{Z}_i$ in Eq. 1. We thus consider a Monte Carlo approximation, sample a group of instances from $p_{\alpha}(z|y_i)$, and calculate the sample mean

$$P_{da}(y|x_i, y_i) \approx \frac{1}{|\hat{\mathcal{Z}}_i|} \sum_{z \in \hat{\mathcal{Z}}_i} P_{\varphi}(y|x_i, z), \quad (2)$$

where $\hat{\mathcal{Z}}_i$ denotes the sampled instances.

There are many possible choices for the latent variable, such as a continuous vector or a categorical discrete variable, which also could be either learned by the model or predefined by rules. Here, we simply represent the latent variable as a sequence of tokens and use predefined rules to generate the sequence, so that the latent variable can be easily incorporated into the input of a seq2seq model without the need for additional parameters.

Specifically, we set the value of the latent variable z to be a group of sampled n-grams from the observed translation y_i and concatenate x_i and z into a sequence of tokens. We assume that the generated translations y can be consistent with the



A. Parallel Data
x_i: <s> die ... diese Einschränkungen für sinnvoll ... </s>
y_i: <s> most ... accept such limits as reasonable ... </s>
Extend
B. Extended Data with Latent z
(x_i, z_1): <s> die ... sinnvoll ... <obs> societies has recently </s>
y_i: <s> most ... accept such limits as reasonable ... </s>
(x_i, z_2): <s> die ... sinnvoll ... <obs> the law as reasonable </s>
y_i: <s> most ... accept such limits as reasonable ... </s>
...
C. Extended Input
(x_i, z_1): <s> die ... sinnvoll ... <obs> societies has recently </s>
(x_i, z_2): <s> die ... sinnvoll ... <obs> the law as reasonable </s>
...
DA model
D. Potential Translations
$\hat{y}_1$: <s> while ... consider these restrictions useful ... </s>
$\hat{y}_2$: <s> most ... regard such restrictions as reasonable ... </s>
...
E. Augmented Data
x_i: <s> die ... diese Einschränkungen für sinnvoll ... </s>
$\hat{y}_1$: <s> while ... consider these restrictions useful ... </s>
x_i: <s> die ... diese Einschränkungen für sinnvoll ... </s>
$\hat{y}_2$: <s> most ... regard such restrictions as reasonable ... </s>
...

Figure 2: The detailed data augmentation process, where the parallel data is augmented multiple times.

observed translation y_i on these n-grams. To this end, we define α as the ratio of tokens in y_i that is observable through z, naming *observed ratio*. For a target with $|y_i|$ tokens, we uniformly sample n-grams from y_i to cover $\alpha \times |y_i|$ tokens that each n-gram has a random length among $\{1, 2, 3\}$. For example, given that $\alpha = 0.1$ and a target y_i with 20 tokens, we can sample one 2-gram or two uni-grams from the target to reach 2 (0.1×20) tokens.

Training. Given a sample (x_i, y_i), the training loss is rewritten as

$$\begin{aligned}\mathcal{L}_{da} &= -\sum_{i=1}^{N} \log P_{da}(y = y_i|x_i, y_i) \\ &\approx -\sum_{i=1}^{N} \log \frac{1}{|\hat{\mathcal{Z}}_i|} \sum_{z \in \hat{\mathcal{Z}}_i} P_{\varphi}(y = y_i|x_i, z) \\ &\leq -\sum_{i=1}^{N} \frac{1}{|\hat{\mathcal{Z}}_i|} \sum_{z \in \hat{\mathcal{Z}}_i} \log P_{\varphi}(y = y_i|x_i, z),\end{aligned} \tag{3}$$

where the upper bound of the loss is provided by Jensen inequality. The upper bound sums log probabilities, which can be seen as sums of the standard negative log-likelihood (NLL) loss of each (x_i, z, y_i). As a result, when we optimize this upper bound as an alternative to optimizing $\mathcal{L}_{da}$, the DA model is trained using standard NLL loss but with $|\hat{\mathcal{Z}}_i|$ times more training instances.

Discussion. As shown in Figure 1, given a sample (x_i, y_i), we adopt a new estimation method using the posterior distribution $P_{da}(y|x_i, y_i)$ for our DA model. The basic intuition is that by conditioning on both the source x_i and the observed translation y_i, the DA model can estimate the data distribution $P_{data}(y|x_i)$ more accurately than an MT model. Logically, an MT model learns a prior distribution $P_{mt}(y|x_i)$, which estimates the data distribution $P_{data}(y|x_i)$ for modeling translation probabilities. This prior distribution works well when the corpus is large. However, when the corpus is sparse in comparison to the data space, the learned distribution overfits the sparsely distributed samples, resulting in poor generalization to unseen targets.

3.2 The Data Augmentation Process

The detailed data augmentation process is shown in Figure 2 and the corresponding algorithm is shown in Algorithm 1. Below we use one training example to illustrate.

DA model training. We represent the latent variable z as a sequence of tokens and concatenate z to the source, so a general seq2seq model can be used to model the posterior distribution. Compared to general MT models, the only difference is the structure of the input.

Specifically, as the step B shown in the figure, for a given sample (x_i, y_i) from the parallel data, we sample a number of n-grams from y_i and extend the input to (x_i, z), where the number is determined according to the length of y_i. Take the target sentence "*most free societies accept such limits as reasonable , but the law has recently become more restrictive .*" as an example. We sample "*societies*" and "*has recently*" from the target and concatenate them to the end of the source sentence to form the first input sequence. We then sample "*the law*" and "*as reasonable*" to form the second input sequence. These new input sequences pair with the original target sequence to form new parallel data. By generating different input sequences, we augment the data multiple times.

Algorithm 1 Target-side data augmentation.

```
Input: $D = \{(x_i, y_i)\}_{i=1}^{N}$                    ▷ A. Parallel data
Output: $D' = \{(x_i, y_i)\}_{i=1}^{N\times(M+1)}$            ▷ Aug $M$ times
function TARGETAUG($D$)
    $D' \leftarrow \{\}$
    for $i \leftarrow 1$ to $N$ do
        $(x_i, y_i) \leftarrow D[i]$                      ▷ For each sample
        $D' \leftarrow D' \cup \{(x_i, y_i)\}$               ▷ Add the gold pair
        for $j \leftarrow 1$ to $M$ do
            $\alpha \sim Beta(a, b)$            ▷ Sample an observed ratio
            $z_j \sim P_\alpha(z|y_i)$                ▷ Sample a latent value
            $\hat{y}_j \sim P_\varphi(y|x_i, z_j)$          ▷ Sample a translation
            $D' \leftarrow D' \cup \{(x_i, \hat{y}_j)\}$          ▷ Add the new pair
    return $D'$                                 ▷ E. Augmented data
```

Target-side data augmentation. Using the data "C. Extended Input" separated from the extended data in step B, we generate new translations by running a beam search with the trained DA model, where for each extended input sequence, we obtain a new translation. Here, we reuse the sampled z from step B. However, we can also sample new z for inference, which does not show an obvious difference in the MT performance. By pairing the new translations with the original source sequence, we obtain "E. Augmented Data". The details are described in Algorithm 1, which inputs the original parallel data and outputs the augmented data.

3.3 The MT Model

We use Transformer (Vaswani et al., 2017) and G-Transformer (Bao et al., 2021) as the baseline MT models. The Transformer baseline models the sentence-level translation and translates a document sentence-by-sentence, while the G-Transformer models the whole document translation and directly translates a source document into the corresponding target document. G-transformer improves the naïve self-attention in Transformer with group-attention (Appendix A) for long document modeling, which is a recent state-of-the-art document MT model.

Baseline Training. The baseline methods are trained on the original training dataset D by the standard NLL loss

$$\mathcal{L}_{mt} = -\sum_{i=1}^{N} \log P_{mt}(y = y_i|x_i). \tag{4}$$

Dataset	Sentences train/dev/test	Documents train/dev/test
TED	0.21M/9K/2.3K	1.7K/92/22
News	0.24M/2K/3K	6K/80/154
Europarl	1.67M/3.6K/5.1K	118K/239/359

Table 1: Datasets statistics.

Augmentation Training. For our target-side augmentation method, we force the MT model to match the posterior distribution estimated by the DA model

$$\mathcal{L}_{mt} = -\sum_{i=1}^{N} \sum_{y\in\mathcal{Y}_i} P_{da}(y|x_i, y_i) \log P_{mt}(y|x_i), \tag{5}$$

where $\mathcal{Y}_i$ is the possible translations of x_i.

We approximate the expectation over $\mathcal{Y}_i$ using a Monte Carlo method. Specifically, for each sample (x_i, y_i), we first sample z_j from $P_\alpha(z|y_i)$ and then run beam search with the DA model by taking x_i and z_j as its input, obtaining a feasible translation. Repeating the process M times, we obtain a set of possible translations

$$\hat{\mathcal{Y}}_i = \{\arg\max_y P_\varphi(y|x_i, z_j)|z_j \sim P_\alpha(z|y_i)\}_{j=1}^{M}, \tag{6}$$

as the step D in Figure 2 and Algorithm 1 in Section 3.2 illustrate.

Subsequently, the loss function for the MT model is rewritten as follows, which approximates the expectation using the average NLL loss of the sampled translations

$$\mathcal{L}_{mt} \approx -\sum_{i=1}^{N} \frac{1}{|\hat{\mathcal{Y}}_i|} \sum_{y\in\hat{\mathcal{Y}}_i} \log P_\theta(y|x_i), \tag{7}$$

where θ denotes the parameters of the MT model. The number $|\hat{\mathcal{Y}}_i|$ could be different for each sample, but for simplicity, we choose a fixed number M in our experiments.

4 Experiments

Datasets. We experiment on three benchmark datasets – TED, News, and Europarl (Maruf et al., 2019), representing different domains and data scales for English-German (En-De) translation. The detailed statistics are displayed in Table 1, and the detailed descriptions are in Appendix B.1.

Metrics. We follow Liu et al. (2020) to use sentence-level BLEU score (s-BLEU) and document-level BLEU score (d-BLEU) as the major metrics for the *performance*. We further define two metrics, including Deviation and Diversity, to measure the quality of generated translations from

Method	TED		News		Europarl		Average
	s-BLEU	d-BLEU	s-BLEU	d-BLEU	s-BLEU	d-BLEU	s-BLEU
HAN (Miculicich et al., 2018)	24.58	-	25.03	-	28.60	-	26.07
SAN (Maruf et al., 2019)	24.42	-	24.84	-	29.75	-	26.34
Hybrid Context (Zheng et al., 2020)	25.10	-	24.91	-	30.40	-	26.80
Flat-Transformer (Ma et al., 2020)	24.87	-	23.55	-	30.09	-	26.17
G-Transformer (rnd.) (Bao et al., 2021)	23.53	25.84	23.55	25.23	32.18	33.87	26.42
G-Transformer (fnt.) (Bao et al., 2021)	25.12	27.17	25.52	27.11	32.39	34.08	27.68
MultiResolution (Sun et al., 2022)	25.24	29.27	25.00	26.71	32.11	34.48	27.45
RecurrentMem (Feng et al., 2022)	25.62	**29.47**	25.73	27.78	31.41	33.50	27.59
SMDT (Zhang et al., 2022)	25.12	-	25.76	-	32.42	-	27.77
Transformer (sent baseline) ◇	24.91	-	24.82	-	31.22	-	26.98
+ Target-side data augmentation (ours)	26.14*	-	27.03*	-	31.75*	-	28.31
G-Transformer (fnt.) (doc baseline) ◇	25.20	27.94	25.12	27.02	31.93	33.88	27.42
+ Target-side augmentation (ours)	**26.59***	29.20*	28.06*	29.83*	**32.85***	**34.76***	**29.17**
Transformer + Back-translation (sent) ♡	25.03	-	26.07	-	31.12	-	27.41
Target-side augmentation (ours)	26.13	-	28.01	-	31.27	-	28.47
G-Transformer + Back-translation (doc) ♡	25.45	28.06	26.25	28.21	32.00	33.94	27.90
Target-side augmentation (ours)	26.21	28.58	**28.69**	**30.41**	32.52	34.50	29.14
Pre-training Setting for Comparison							
Flat-Transformer+BERT (Ma et al., 2020)	26.61	-	24.52	-	31.99	-	27.71
G-Transformer+BERT (Bao et al., 2021)	26.81	-	26.14	-	32.46	-	28.47
G-Transformer+mBART (Bao et al., 2021)	28.06	30.03	30.34	31.71	32.74	34.31	30.38

Table 2: Main results evaluated on English-German document-level translation, where "*" indicates a significant improvement upon the baseline with $p < 0.01$. (rnd.) – parameters are randomly initialized. (fnt.) – parameters are initialized using a trained sentence model. ◇ – we adjust the hyper-parameters for augmented datasets. ♡ – we augment the training data by back-translating each target to a new source instead of introducing additional monolingual targets.

the DA model for *analysis*. The detailed description and definition are in Appendix B.2.

Baselines. We apply target-side augmentation to two baselines, including sentence-level Transformer (Vaswani et al., 2017) and document-level G-transformer (Bao et al., 2021). We further combine back-translation and target-side augmentation, and apply it to the two baselines.

Training Settings. For both Transformer and G-Transformer, we generate M new translations (9 for TED and News, and 3 for Europarl) for each sentence and augment the data to its $M + 1$ times. For back-translation baselines, where the training data have already been doubled, we further augment the data 4 times for TED and News, and 1 for Europarl, so that the total times are still 10 for TED and News, and 4 for Europarl.

We obtain the translations by sampling latent z with an observed ratio from a Beta distribution $Beta(2, 3)$ and running a beam search with a beam size of 5. We run each main experiment three times and report the median. More details are described in Appendix B.3.

4.1 Main Results

As shown in Table 2, target-side augmentation significantly improves all the *baselines*. Particularly, it improves G-Transformer (fnt.) by 1.75 s-BLEU on average over the three benchmarks, where the improvement on News reaches 2.94 s-BLEU. With the augmented data generated by the DA model, the gap between G-Transformer (rnd.) and G-Transformer (fnt.) narrows from 1.26 s-BLEU on average to 0.18, suggesting that fine-tuning on sentence MT model might not be necessary when augmented data is used. For the Transformer baseline, target-side augmentation enhances the performance by 1.33 s-BLEU on average. These results demonstrate that target-side augmentation can significantly improve the baseline models, especially on small datasets.

Comparing with *previous work*, G-Transformer (fnt.)+Target-side augmentation outperforms the best systems SMDT, which references retrieved similar translations, with a margin of 1.40 s-BLEU on average. It outperforms previous competitive RecurrentMem, which gives the best score on TED, with a margin of 1.58 s-BLEU on average. Compared with MultiResolution, which is also a data augmentation approach that increases the training data by splitting the documents into different resolutions (e.g., 1, 2, 4, 8 sentences per training instance), target-side augmentation obtains higher performance with a margin of 1.72 s-BLEU on average. With target-side augmentation, G-Transformer (fnt.) achieves the best-reported s-BLEU on all

Method	TED	News	Europarl	Increase
G-Transformer (fnt.)	25.20	25.12	31.93	-
+ Prior-based aug	25.69	26.34	32.16	+0.64
+ Posterior-based aug	26.59	28.06	32.85	+1.75

Table 3: MT performance with prior/posterior-based DA models, evaluated in *s-BLEU*.

three datasets.

Compared to the *pre-training setting*, target-side augmentation with G-Transformer (fnt.) outperforms Flat-Transformer+BERT and G-Transformer+BERT, which are fine-tuned on pre-trained BERT, with margins of 1.46 and 0.70 s-BLEU, respectively, on an average of the three benchmarks, where the margins on News reaches 3.54 and 1.92, respectively. The score on bigger dataset Europarl even excels strong large pre-training G-Transformer+mBART, suggesting the effectiveness of target-side augmentation for both small and large datasets.

Back-translation does not enhance the performance on TED and Europarl by an adequate margin, but enhances the performance on News significantly, compared to the Transformer and G-Transformer baselines. Upon the enhanced baselines, target-side augmentation further improves the performance on News to a new level, reaching the highest s/d-BLEU scores of 28.69 and 30.41, respectively. The results demonstrate that target-side augmentation complements the back-translation technique, where a combination may be the best choice in practice.

4.2 Posterior vs Prior Distribution

We first compare the MT performance of using a posterior distribution $P(y|x_i, y_i)$ in the DA model (Eq. 5 in Section 3.3) against using the prior distribution $P(y|x_i)$. As shown in Table 3, when using a prior-based augmentation, the performance improves by 0.64 s-BLEU on average compared to using the original data. After replacing the DA model with the posterior distribution, the performance improves by 1.75 s-BLEU on average, which is larger than the improvements obtained by the prior distribution. The results suggest that using a DA model (even with a simple prior distribution) to augment the target sequence is effective, and the posterior distribution further gives a significant boost.

Generated Translations. We evaluate the distribution of generated translations, as shown in Table 4. Using prior distribution, we obtain translations with higher Diversity than posterior distribution.

Method	Diversity ↑	Deviation ↓	PPL ↓
Prior distribution	**78.68**	76.55	8.68
Posterior distribution	45.42	**47.14**	**7.00**

Table 4: Quality of generated translations and accuracy of the estimated distributions from the DA model, evaluated on *News*.

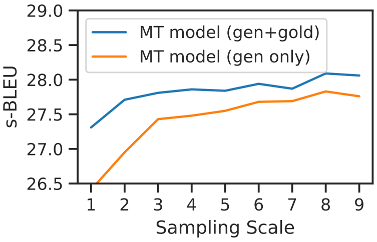


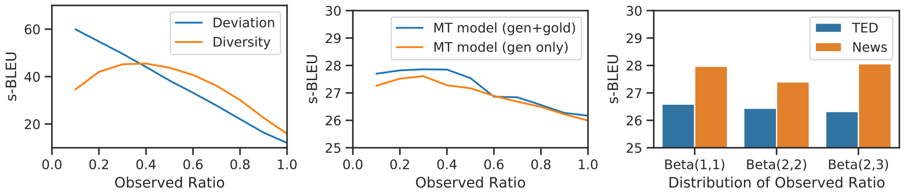
Figure 3: Impact of the sampling scale for z, trained on G-Transformer (fnt.) and evaluated in *s-BLEU* on *News*. (gen+gold) – trained on both generated and gold translations. (gen only) – trained on generated translations.

However, higher Diversity does not necessarily lead to better performance if the generated translations are not consistent with the target distribution. As the Deviation column shows, the translations sampled from the posterior distribution have a much smaller Deviation than that from the prior distribution, which confirms that the DA model estimating posterior distribution can generate translations more similar to the gold target.

Accuracy of Estimated Distribution. As more direct evidence to support the DA model with a posterior distribution, we evaluate the perplexity (PPL) of the model on a multiple-reference dataset, where a better model is expected to give a lower PPL on the references (Appendix C.1). As shown in the column PPL in Table 4, we obtain an average PPL (per token) of 7.00 for the posterior and 8.68 for the prior distribution, with the former being 19.4% lower than the latter, confirming our hypothesis that the posterior distribution can estimate the data distribution $P_{data}(y|x_i)$ more accurately.

4.3 Sampling of Latent z

Scale. The sampling scale $|\hat{\mathcal{Y}}|$ in Eq. 7 is an important influence factor on the model performance. Theoretically, the larger the scale is, the more accurate the approximation will be. Figure 3 evaluates the performance on different scales of generated translations. The overall trends confirm the theoretical expectation that the performance improves when the scale increases. At the same time, the contribution of the gold translation drops when the scale increases, suggesting that with more generated translations, the gold translation provides



(a) Quality of translations generated by the DA model, evaluated on *News*

(b) Performance of MT model on augmented data, evaluated on *News*

(c) Performance of MT models trained using mixed observed ratios

Figure 4: Impact of the observed ratio for z, trained on G-Transformer (fnt.) and evaluated in *s-BLEU*. Beta(a,b) – The function curves are shown in Appendix B.3.

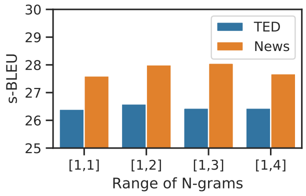


Figure 5: Impact of the granularity of n-grams, trained on G-Transformer (fnt.) and evaluated in *s-BLEU*.

less additional information. In addition, the performance of scale ×1 and ×9 have a gap of 0.75 s-BLEU, suggesting that the MT model requires sufficient samples from the DA model to match its distribution. In practice, we need to balance the performance gain and the training costs to decide on a suitable sampling scale.

Observed Ratio. Using the observed ratio (α in Eq. 1), we can control the amount of information provided by the latent variable z. Such a ratio influences the quality of generated translations. As Figure 4a shows, a higher observed ratio produces translations with a lower Deviation from the gold reference, which shows a monotonic descent curve. In comparison, the diversity of the generated translations shows a convex curve, which has low values when the observed ratio is small or big but high values in the middle. The diversity of the generated translations represents the degree of smoothness of the augmented dataset, which has a direct influence on the model performance.

As Figure 4b shows, the MT model obtains the best performance around the ratio of 0.4, where it has a balanced quality of Deviation and Diversity. When the ratio further increases, the performance goes down. Comparing the MT models trained with/without the gold translation, we see that the performance gap between the two settings is closing when the observed ratio is bigger than 0.6, where the generated translations have low Deviation from the gold translations.

The Diversity can be further enhanced by mixing the generated translations from different observed ratios. Therefore, instead of using a fixed ratio, we sample the ratio from a predefined Beta distribution. As Figure 4c shows, we compare the performance on different Beta distributions. The performance on TED peaks at $Beta(1, 1)$ but does not show a significant difference compared to the other two, while the performance on News peaks at $Beta(2, 3)$, which has a unimodal distribution with an extremum between the ratio 0.3 and 0.4 and has a similar shape as the curve of Diversity in Figure 4a. Compared to $Beta(2, 2)$, which is also a unimodal distribution but with an extremum at the ratio 0.5, the performance with $Beta(2, 3)$ is higher by 0.66 s-BLEU.

Method	TED	News	Increase
	s/d-BLEU	s/d-BLEU	s-BLEU
G-Transformer (fnt.)	25.20 / 27.94	25.12 / 27.02	-
+ Source-side aug	25.74 / 28.30	26.82 / 28.61	+1.12
+ Target-side aug	26.59 / 29.20	28.06 / 29.83	+2.17
+ Both-side aug	26.85 / 29.46	28.31 / 29.99	+2.42

Table 5: Source-side vs. target-side augmentations.

Granularity of N-grams. The granularity of n-grams determines how much order information between tokens is observable through the latent z (in comparison, the observed ratio determines how many tokens are observed). We evaluate different ranges of n-grams, where we sample n-grams according to a number uniformly sampled from the range. As Figure 5 shows, the performance peaks at $[1, 2]$ for TED and $[1, 3]$ for News. However, the differences are relatively small, showing that the performance is not sensitive to the token order of the original reference. A possible reason may be that the DA model can reconstruct the order according to the semantic information provided by the source sentence.

4.4 Different Augmentation Methods

Source-side and Both-side Augmentation. We compare target-side augmentation with the source-side and both-side augmentations, by applying the DA model to the source and both sides. As Table 5 shows, the source-side augmentation improves the baseline by 1.12 s-BLEU on average of TED and News but is still significantly lower than the target-side augmentation, which improves the baseline by 2.17 s-BLEU on average. Combining the

Method	Dev	Test
Transformer (base)	34.85	33.87
+ T5 paraphraser ◇	34.01	33.10
+ Target-side augmentation	**36.42**	**35.42**

Table 6: Target-side augmentation vs paraphraser on sentence-level MT, evaluated on IWSLT14 German-English (De-En). ◇ – nucleus sampling with $p = 0.95$.

generated data from both the source-side and target-side augmentations, we obtain an improvement of 2.42 s-BLEU on average, whereas the source-side augmented data further enhance the target-side augmentation by 0.25 s-BLEU on average. These results suggest that the DA model is effective for source-side augmentation but more significantly for target-side augmentation.

Paraphrasing. Target-side augmentation augments the parallel data with new translations, which can be seen as paraphrases of the original gold translation. Such paraphrasing can also be achieved by external paraphrasers. We compare target-side augmentation with a pre-trained T5 paraphraser on a sentence-level MT task, using the settings described in Appendix C.3.

As shown in Table 6, the T5 paraphraser performs lower than the Transformer baseline on both the dev and test sets, while target-side augmentation outperforms the baseline by 1.57 and 1.55 on dev and test, respectively. The results demonstrate that a DA model is effective for sentence MT but a paraphraser may not, which can be because of missing translation information.

In particular, the generated paraphrases from the T5 paraphraser have a Diversity of 40.24, which is close to the Diversity of 37.30 from the DA model. However, when we compare the translations by calculating the perplexity (PPL) on the baseline Transformer, we get a PPL of 3.40 for the T5 paraphraser but 1.89 for the DA model. The results suggest that compared to an external paraphraser, the DA model generates translations more consistent with the distribution of the gold targets.

4.5 Further Analysis

Size of The DA model. The condition on an observed translation simplifies the DA model for predicting the target. As a result, the generated translations are less sensitive to the capacity of the DA model. Results with different sizes of DA models confirm the hypothesis and suggest that the MT performance improves even with much smaller DA models. The details are in Appendix C.2.

Case Study. We list several word, phrase, and sentence cases of German-English translations, and two documents of English-German translations, demonstrating the diversity of the generated translations by the DA model. The details are shown in Appendix C.4.

5 Conclusion

We investigated a target-side data augmentation method, which introduces a DA model to generate many possible translations and trains an MT model on these smoothed targets. Experiments show our target-side augmentation method reduces the effect of data sparsity issues, achieving strong improvement upon the baselines and new state-of-the-art results on News and Europarl. Analysis suggests that a balance between high Diversity and low Deviation is the key to the improvements. To our knowledge, we are the first to do target-side augmentation in the context of document-level MT.

Limitations

Long documents, intuitively, have more possible translations than short documents, so a dynamic number of generated translations may be a better choice when augmenting the data, which balances the training cost and the performance gain. Another potential solution is to sample a few translations and force the MT model to match the dynamic distribution of the DA model using these translations as decoder input, similar to Khayrallah et al. (2020). Such dynamic sampling and matching could potentially be used to increase training efficiency. We do not investigate the solution in this paper and leave the exploration of this topic to future work.

Target-side augmentation can potentially be applied to other seq2seq tasks, where the data sparsity is a problem. Due to the limitation of space in a conference submission, we will leave investigations on other tasks for future work.

Acknowledgements

We would like to thank the anonymous reviewers for their valuable feedback. This work is funded by the China Strategic Scientific and Technological Innovation Cooperation Project (grant No. SQ2022YFE020038) and the National Natural Science Foundation of China (grant NSFC No. 62161160339). Zhiyang Teng is partially supported by CAAI-Huawei MindSpore Open Fund (CAAIXSJLJJ-2021-046A).

References

Jimmy Ba and Rich Caruana. 2014. Do deep nets really need to be deep? *Advances in neural information processing systems*, 27.

Guangsheng Bao, Zebin Ou, and Yue Zhang. 2023. Gemini: Controlling the sentence-level writing style for abstractive text summarization. *arXiv preprint arXiv:2304.03548*.

Guangsheng Bao and Yue Zhang. 2021. Contextualized rewriting for text summarization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 12544–12553.

Guangsheng Bao and Yue Zhang. 2023. A general contextualized rewriting framework for text summarization. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*.

Guangsheng Bao, Yue Zhang, Zhiyang Teng, Boxing Chen, and Weihua Luo. 2021. G-transformer for document-level 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 3442–3455.

Mikhail Belkin, Daniel Hsu, Siyuan Ma, and Soumik Mandal. 2019. Reconciling modern machine-learning practice and the classical bias–variance trade-off. *Proceedings of the National Academy of Sciences*, 116(32):15849–15854.

Nikolay Bogoychev and Rico Sennrich. 2019. Domain, translationese and noise in synthetic data for neural machine translation. *arXiv preprint arXiv:1911.03362*.

Verna Dankers, Elia Bruni, and Dieuwke Hupkes. 2022. The paradox of the compositionality of natural language: A neural machine translation case study. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4154–4175.

Marzieh Fadaee, Arianna Bisazza, and Christof Monz. 2017. Data augmentation for low-resource neural machine translation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 567–573.

Akhbardeh Farhad, Arkhangorodsky Arkady, Biesialska Magdalena, Bojar Ondřej, Chatterjee Rajen, Chaudhary Vishrav, Marta R Costa-jussa, España-Bonet Cristina, Fan Angela, Federmann Christian, et al. 2021. Findings of the 2021 conference on machine translation (wmt21). In *Sixth Conference on Machine Translation*, pages 1–88. Association for Computational Linguistics.

Yukun Feng, Feng Li, Ziang Song, Boyuan Zheng, and Philipp Koehn. 2022. Learn to remember: Transformer with recurrent memory for document-level machine translation. *arXiv preprint arXiv:2205.01546*.

Jianfeng Gao, Xiaodong He, Wen-tau Yih, and Li Deng. 2014. Learning continuous phrase representations for translation modeling. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 699–709.

Zhengxian Gong, Min Zhang, and Guodong Zhou. 2011. Cache-based document-level statistical machine translation. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 909–919.

Mitchell A Gordon and Kevin Duh. 2019. Explaining sequence-level knowledge distillation as data-augmentation for neural machine translation. *arXiv preprint arXiv:1912.03334*.

Jianping Gou, Baosheng Yu, Stephen J Maybank, and Dacheng Tao. 2021. Knowledge distillation: A survey. *International Journal of Computer Vision*, 129(6):1789–1819.

Christian Hardmeier, Sara Stymne, Jörg Tiedemann, and Joakim Nivre. 2013. Docent: A document-level decoder for phrase-based statistical machine translation. In *Acl 2013 (51st annual meeting of the association for computational linguistics); 4-9 august 2013; sofia, bulgaria*, pages 193–198. Association for Computational Linguistics.

Junxian He, Jiatao Gu, Jiajun Shen, and Marc'Aurelio Ranzato. 2019. Revisiting self-training for neural sequence generation. In *International Conference on Learning Representations*.

Xuanli He, Gholamreza Haffari, and Mohammad Norouzi. 2018. Sequence to sequence mixture model for diverse machine translation. In *Proceedings of the 22nd Conference on Computational Natural Language Learning*, pages 583–592.

Geoffrey Hinton, Oriol Vinyals, Jeff Dean, et al. Distilling the knowledge in a neural network.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2019. The curious case of neural text degeneration. In *International Conference on Learning Representations*.

J Edward Hu, Huda Khayrallah, Ryan Culkin, Patrick Xia, Tongfei Chen, Matt Post, and Benjamin Van Durme. 2019. Improved lexically constrained decoding for translation and monolingual rewriting. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 839–850.

Marcin Junczys-Dowmunt. 2019. Microsoft translator at wmt 2019: Towards large-scale document-level neural machine translation. In *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, pages 225–233.

Huda Khayrallah, Brian Thompson, Matt Post, and Philipp Koehn. 2020. Simulated multiple reference training improves low-resource machine translation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 82–89.

Yoon Kim and Alexander M Rush. 2016. Sequence-level knowledge distillation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1317–1327.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, et al. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th annual meeting of the association for computational linguistics companion volume proceedings of the demo and poster sessions*, pages 177–180.

Philipp Koehn and Rebecca Knowles. 2017. Six challenges for neural machine translation. In *First Workshop on Neural Machine Translation*, pages 28–39. Association for Computational Linguistics.

Taku Kudo. 2018. Subword regularization: Improving neural network translation models with multiple subword candidates. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 66–75.

Bohan Li, Yutai Hou, and Wanxiang Che. 2022. Data augmentation approaches in natural language processing: A survey. *AI Open*.

Yafu Li, Yongjing Yin, Yulong Chen, and Yue Zhang. 2021. On compositional generalization of 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 4767–4780.

Alexander Lin, Jeremy Wohlwend, Howard Chen, and Tao Lei. 2020. Autoregressive knowledge distillation through imitation learning. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6121–6133.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020. Multilingual denoising pre-training for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:726–742.

Shuming Ma, Dongdong Zhang, and Ming Zhou. 2020. A simple and effective unified encoder for document-level machine translation. In *Proceedings of the 58th annual meeting of the association for computational linguistics*, pages 3505–3511.

Nitin Madnani, Necip Fazil Ayan, Philip Resnik, and Bonnie Dorr. 2007. Using paraphrases for parameter tuning in statistical machine translation. In *Proceedings of the Second Workshop on Statistical Machine Translation*, pages 120–127.

Sameen Maruf, André FT Martins, and Gholamreza Haffari. 2019. Selective attention for context-aware neural machine translation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3092–3102.

Lesly Miculicich, Dhananjay Ram, Nikolaos Pappas, and James Henderson. 2018. Document level neural machine translation with hierarchical attention networks. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, CONF.

Mohammad Norouzi, Samy Bengio, Navdeep Jaitly, Mike Schuster, Yonghui Wu, Dale Schuurmans, et al. 2016. Reward augmented maximum likelihood for neural structured prediction. *Advances In Neural Information Processing Systems*, 29.

Artidoro Pagnoni, Kevin Liu, and Shangyan Li. 2018. Conditional variational autoencoder for neural machine translation. *arXiv preprint arXiv:1812.04405*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

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.

Beatrice Savoldi, Marco Gaido, Luisa Bentivogli, Matteo Negri, and Marco Turchi. 2021. Gender bias in machine translation. *Transactions of the Association for Computational Linguistics*, 9:845–874.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016a. Improving neural machine translation models with monolingual data. In *54th Annual Meeting of the Association for Computational Linguistics*, pages 86–96. Association for Computational Linguistics (ACL).

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016b. Neural machine translation of rare words with subword units. In *54th Annual Meeting of the Association for Computational Linguistics*, pages 1715–1725. Association for Computational Linguistics (ACL).

Connor Shorten and Taghi M Khoshgoftaar. 2019. A survey on image data augmentation for deep learning. *Journal of big data*, 6(1):1–48.

Connor Shorten, Taghi M Khoshgoftaar, and Borko Furht. 2021. Text data augmentation for deep learning. *Journal of big Data*, 8(1):1–34.

Zewei Sun, Mingxuan Wang, Hao Zhou, Chengqi Zhao, Shujian Huang, Jiajun Chen, and Lei Li. 2022. Rethinking document-level neural machine translation. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 3537–3548.

Brian Thompson and Matt Post. 2020. Paraphrase generation as zero-shot multilingual translation: Disentangling semantic similarity from lexical and syntactic diversity. In *Proceedings of the Fifth Conference on Machine Translation*, pages 561–570.

David A Van Dyk and Xiao-Li Meng. 2001. The art of data augmentation. *Journal of Computational and Graphical Statistics*, 10(1):1–50.

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.

Michelle Y Wang and Trevor Park. 2020. A brief tour of bayesian sampling methods. *Bayesian inference on complicated data*, 17.

Xinyi Wang, Hieu Pham, Zihang Dai, and Graham Neubig. 2018. Switchout: an efficient data augmentation algorithm for neural machine translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 856–861.

Jason Wei and Kai Zou. 2019. Eda: Easy data augmentation techniques for boosting performance on text classification tasks. 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 6382–6388.

Lesly Miculicich Werlen, Dhananjay Ram, Nikolaos Pappas, and James Henderson. 2018. Document-level neural machine translation with hierarchical attention networks. In *EMNLP*.

Xing Wu, Shangwen Lv, Liangjun Zang, Jizhong Han, and Songlin Hu. 2019. Conditional bert contextual augmentation. In *International conference on computational science*, pages 84–95. Springer.

Shufang Xie, Ang Lv, Yingce Xia, Lijun Wu, Tao Qin, Tie-Yan Liu, and Rui Yan. 2022. Target-side input augmentation for sequence to sequence generation. In *International Conference on Learning Representations*.

Biao Zhang, Deyi Xiong, Jinsong Su, Hong Duan, and Min Zhang. 2016. Variational neural machine translation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 521–530.

Xu Zhang, Jian Yang, Haoyang Huang, Shuming Ma, Dongdong Zhang, Jinlong Li, and Furu Wei. 2022. Smdt: Selective memory-augmented neural document translation. *arXiv preprint arXiv:2201.01631*.

Yuan Zhang, Jason Baldridge, and Luheng He. 2019. Paws: Paraphrase adversaries from word scrambling. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1298–1308.

Zaixiang Zheng, Yue Xiang, Shujian Huang, Jiajun Chen, and Alexandra Birch-Mayne. 2020. Toward making the most of context in neural machine translation. In *29th International Joint Conference in Artificial Intelligence*, pages 3983–3989. International Joint Conferences on Artificial Intelligence Organization.

Barret Zoph, Golnaz Ghiasi, Tsung-Yi Lin, Yin Cui, Hanxiao Liu, Ekin Dogus Cubuk, and Quoc Le. 2020. Rethinking pre-training and self-training. *Advances in neural information processing systems*, 33:3833–3845.

A G-Transformer

G-Transformer (Bao et al., 2021) has an encoder-decoder architecture, involving two types of multi-head attention. One is for global document, naming *global attention*, while another is for local sentence, naming *group attention*.

Global Attention. The global attention is simply a normal multi-head attention, which attends to the whole document.

$$\begin{aligned} args &= (Q, K, V), \\ \text{GlobalAttn}(args) &= \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V, \end{aligned} \tag{8}$$

where matrix inputs Q, K, V are query, key, and value for calculating the attention.

Group Attention. The group attention differentiates the sentences in a document by assigning a group tag (Bao and Zhang, 2021, 2023; Bao et al., 2023) to each sentence. The group tag is a number used to identify a specific sentence, which is allocated in the order of sentences, where the group tag for the first sentence is 1, second sentence 2, and so on.

The group tag sequences are used to calculate an attention mask to avoid cross-sentential attention

$$\begin{aligned} args &= (Q, K, V, G_Q, G_K), \\ \text{GroupAttn}(args) &= \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}} + M(G_Q, G_K)\right)V, \end{aligned} \tag{9}$$

where G_Q and G_K are group-tag sequences for query and key. The function $M(G_Q, G_K)$ calculates the attention mask that for a group tag in G_Q and a group tag in G_K, it returns a big negative number if the two tags are different, otherwise it returns 0.

Combined Attention The two multi-head attentions are combined using a gate-sum module

$$\begin{aligned} H_L &= \text{GroupMHA}(Q, K, V, G_Q, G_K), \\ H_G &= \text{GlobalMHA}(Q, K, V), \\ g &= \text{sigmoid}([H_L, H_G]W + b), \\ H &= H_L \odot g + H_G \odot (1 - g), \end{aligned} \tag{10}$$

where W and b are trainable parameters, and $\odot$ denotes element-wise multiplication.

G-Transformer uses group attention on low layers and combined attention on top 2 layers.

B Datasets and Metrics

B.1 Datasets

The three benchmark datasets are as follows.

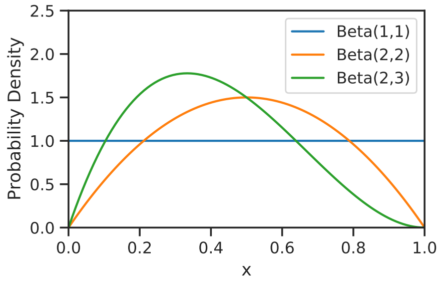


Figure 6: The probability density function of $Beta(a, b)$ distributions.

TED is a corpus from IWSLT2017, which contains the transcriptions of TED talks that each talk corresponds to a document. The sentences in source and target documents are aligned for translation. We use *tst2016-2017* for testing and the rest for development.

News is a corpus mainly from News Commentary v11, where the sentences are also aligned between the source and target documents. We use *newstest2016* for testing and *newstest2015* for development. In addition, we use *newstest2021* from WMT21 (Farhad et al., 2021), which has three references for each source, to evaluate the quality of the estimation of data distribution.

Europarl is a corpus extracted from Europarl v7, where the train, development, and test sets are randomly split.

We pre-process the data by tokenizing and truecasing the sentences using MOSES tools (Koehn et al., 2007), followed with a BPE (Sennrich et al., 2016b) of 30000 merging operations.

B.2 Metrics

The sentence-level BLEU score (s-BLEU) and document-level BLEU score (d-BLEU) are described as follows.

s-BLEU is calculated over sentence pairs between the source and target document, which is basically the same with the BLEU scores (Papineni et al., 2002) for sentence NMT models.

d-BLEU is calculated over document pairs, taking each document as a whole word sequence and computing the BLEU scores between the source and target sequences.

For *analysis*, we measure the Deviation and Diversity of generated translations.

Deviation is simply defined as the distance to perfect s-BLEU score

$$\text{Deviation}(\hat{y}, y) = 100 - \text{s-BLEU}(\hat{y}, y), \tag{11}$$

Teacher Size	Hyperparameters	Params	TED	News	Europarl	Average
Base	6 layers, 8 heads, hidden 512, FFN 2048	69M	26.59	28.06	32.85	29.17
Small	6 layers, **4** heads, hidden 512, FFN **1024**	56M	26.64	28.23	32.42	29.10
Tiny	6 layers, **4** heads, hidden **256**, FFN **1024**	21M	26.73	28.08	32.30	29.04

Table 7: Impact of the size of the DA model, trained on G-Transformer (fnt.) and evaluated in *s-BLEU*.

where $\hat{y}$ is a generated translation and y is the reference translation.

Diversity is calculated by averaging the deviation scores among the generated translations

$$\text{Diversity}(\hat{\mathcal{Y}}) = \frac{\sum_{i=1}^{M}\sum_{j=i+1}^{M}\text{Deviation}(\hat{y}_i, \hat{y}_j)}{M(M-1)/2}, \quad (12)$$

where $\hat{\mathcal{Y}}$ is a set of generated translations, containing M elements. The metric is similar to a diversity metric in He et al. (2018) beside that we use s-BLEU for basic measure.

B.3 Training Settings

We use a base model for all the baselines, where the models have around 60M parameters. We adjust several hyper-parameters of the default setting to better suit the augmented data. First, we extend the maximum length of the model from 512 to 1024. Next, we change the dropout from 0.3 to 0.1 for Europarl but keep the dropout of 0.3 for News and TED. Last, we reduce the patience of training the DA model from 10 to 5 for TED and News, and from 10 to 2 for Europarl, so that the training process could be accelerated.

Running with the new settings on 4 Tesla V100 GPUs, the G-Transformer (fnt.) baseline takes 2 hours to train on TED, 2.5 hours on News, and 13 hours on Europarl. After augmenting the data 9 times, the training of G-Transformer (fnt.) for the MT model costs 10, 16, and 49 hours on TED, News, and Europarl, respectively. In comparison, the training for the DA model costs 5, 8.5, and 25 hours on TED, News, and Europarl, respectively.

Beta Distributions. We use a Beta distribution to sample the observed ratio, where we consider three basic candidates including $Beta(1, 1)$, $Beta(2, 2)$, and $Beta(2, 3)$ as Figure 6 displays.

We decide on the choice by comparing the figure to the Diversity curve shown in Figure 4a, where we can see that $Beta(2, 3)$ has the best match with the Diversity curve of the generated translations. Our further analysis in Section 4.3 confirms that $Beta(2, 3)$ provides a balanced performance on TED and News.

C More Analysis

C.1 Multi-reference Evaluation

As more direct evidence that a DA model with a posterior distribution estimates $P_{data}(y|x_i)$ more accurately than that with a prior distribution, we evaluate the perplexity (PPL) on a multiple-reference dataset *newstest2021*, which contains 67 documents and 1002 source sentences, each with 3 translations. We cross-validate the translations by using one as an observed translation and the other two as test translations. Using Eq. 2, we approximate the posterior probability by sampling the latent z sufficient times (e.g., 100).

C.2 Size of The DA Model

The posterior distribution simplifies the translation task for the DA model since the input latent z contains much information about the target. As a result, the DA model is less sensitive to the capacity of the model. We evaluate target-side augmentation with different sizes of DA models. The results are shown in Table 7. The performance on TED and News does not show a significant difference when we reduce the number of parameters from 69M to 21M. On bigger Europarl, the performance drops by 0.55 s-BLEU but still outperforms the baseline G-Transformer (fnt.) by 0.37 s-BLEU, suggesting that the DA model provides additional value even when its capacity is much lower than the MT model.

C.3 Paraphrasing Settings

We use the T5 paraphraser [1], created by fine-tuning T5 (Raffel et al., 2020) on English paraphrases (Zhang et al., 2019), as a representative to make a comparative study. Given that the T5 paraphraser is trained in English and works at the sentence level, we translate the documents sentence-by-sentence and evaluate the methods on MT benchmark IWSLT14 German-English. For each target sentence, we sample 6 paraphrases by running nucleus sampling (Holtzman et al., 2019) with the T5 paraphraser. For target-side augmentation, we

[1] https://huggingface.co/Vamsi/T5_Paraphrase_Paws

Level	Source	Target	Generated Translations
Word	herauszufinden	identify	find out, figure out, find, learn, look out, see
	unglaublich	incredibly	unbelievable, amazingly, extremely, highly, remarkably
	überzeugt	convinced	persuaded, believed, pretty sure
Phrase	halten diese Einschränkungen für sinnvoll	accept such limits as reasonable	1) *consider these restrictions useful* 2) *regard such restrictions as reasonable* 3) *take these constraints as certain*
	passiv bewegte ohren sobald der kopf etwas tut .	ears that move passively when the head goes .	1) *ears moving passively* when the head does something . 2) *passively moving ears* once the head goes . 3) *passive ears that move* when your head does something .
	ein aus holz und stoff gebautes objekt ist , mit eingebauten bewegungen , um euch glauben zu lassen , sie sei lebendig .	an object constructed out of wood and cloth with movement built into it to persuade you to believe that it has life	1) *an object made out of* wood and cloth , with movement built in to persuade you to believe that *it 's alive* . 2) an object built out of wood and cloth *with movement to perpetuate you to believe it 's alive* . 3) *a wooden and cloth object* with movement built in to make you believe that it 's alive .
Sentence	sie lebt nur dann wenn man sie dazu bringt .	it only lives because you make it .	1) it only lives *when you get it to do* . 2) it lives *only as you make it* . 3) it only lives *because you get them to do it* .
	in jedem moment auf der bühne rackert sich die puppe ab .	so every moment it 's on the stage , it 's making the struggle .	1) *at every moment on the stage* , it 's making the struggle of puppet . 2) every moment on the stage *it reckers down the puppet* . 3) so every moment it 's on the stage , *the puppet is racking off* .
	er demonstriert anhand einer schockierenden geschichte von der toxinbelastung auf einem japanischen fischmarkt , wie gifte den weg vom anfang der ozeanischen nahrungskette bis in unseren körper finden .	he shows how toxins at the bottom of the ocean food chain find their way into our bodies , with a shocking story of toxic contamination from a japanese fish market .	1) *he demos through a shocking story of toxic burden* on a japanese fish market , *how poisoning their way from the beginning of the ocean food chain into our bodies* . 2) he demos through a shocking story of toxin impact on a japanese fish market , *how poised the way from the ocean food chain to our bodies* . 3) he demos through a shocking story of toxin contamination at a japanese fish market , *with how toxins find the way from the beginning of the ocean food chain to our bodies* .

Table 8: Translations generated by the DA model on IWSLT14 German-English.

generate 6 translations for each source sentence without using the document context. It is worth noting that different from the previous paraphrasing augmentation method (Khayrallah et al., 2020), where the MT model learns the paraphraser's distribution directly, we use sampled text output to train the MT models.

C.4 Case Study

Our case study demonstrates that the DA model generates diverse translations at word, phrase, and sentence levels. Several cases for German-English translation are listed in Table 8.

We further list two document-level translations, through which we can have a direct sense of how target-side augmentation improves MT performance, as Table 9 shows.

Source: Elton John and Russian President Vladimir Putin to meet to discuss gay rights in 2003, Mikhail Khodorkovsky, Russia 's wealthiest man, was arrested at gunpoint on a Siberian runway. having openly challenged President Vladimir Putin, Khodorkovsky was convicted, his oil company, Yukos, seized and his pro democracy efforts curtailed.

Target: Elton John und der russische Präsident Vladimir Putin treffen sich, um Rechte der Schwulen zu diskutieren Mikhail Khodorkovsky, *Russlands reichster Mann*, wurde auf einem sibirischen Rollfeld mit Waffengewalt verhaftet. nachdem er Präsident Vladimir Putin offen herausgefordert hatte, wurde Khodorkovsky verurteilt, sein Ölunternehmen Yukos beschlagnahmt und *seine demokratischen Bemühungen unterbunden*.

Baseline: Elton John und der russische Präsident Wladimir Putin müssen sich treffen, um über Homosexuelle zu diskutieren im Jahr 2003 wurde Michail Chodorkowski, *der reichste Mann Russlands*, an einer sibirischen Stichwahl verhaftet. nachdem er Präsident Wladimir Putin offen in Frage gestellt hatte, wurde Chodorkowski verurteilt, seine Ölgesellschaft Yukos, beschlagnahmt und *seine Anstrengungen zur Demokratie beschnitten*.

Ours: Elton John und der russische Präsident Wladimir Putin treffen sich, um über Homosexuellenrechte zu diskutieren 2003 wurde Michail Chodorkowski, *Russlands reichster Mann*, auf einer sibirischen Stichwahl verhaftet. nachdem er Präsident Wladimir Putin offen in Frage gestellt hatte, wurde Chodorkowski verurteilt, seine Ölgesellschaft Yukos erobert und *seine Bemühungen zur Demokratie eingeschränkt*.

Source: the Upper Bavarian district of Ramsau bei Berchtesgaden is Germany 's first "Mountaineers 'Village". the village of 1,800 inhabitants in the Berchtesgaden National Park received the award for "gentle Tourism" from the hand of the Vice President of the German Alpine Association, Ludwig Wucherpfenning, on Wednesday. there are already 20 "Mountaineers' Villages" in Austria. in our neighbouring country, the local Alpine Association is responsible for awarding the distinction. a "Mountaineers 'Village" is permitted to have a maximum of 2,500 residents. at least one fifth of its area must be designated as a protected area.

Target: die oberbayerische Gemeinde Ramsau bei Berchtesgaden ist Deutschlands erstes "Bergsteigerdorf". *aus der Hand des Vizepräsidenten* beim Deutschen Alpenverein, Ludwig Wucherpfennig, erhielt das 1800-Einwohner-Dorf im Nationalpark Berchtesgaden am Mittwoch die Auszeichnung für sanften Tourismus. in Österreich gibt es bereits 20 "Bergsteigerdörfer". im Nachbarland ist der dortige Alpenverein für die Vergabe der Auszeichnung zuständig. ein "Bergsteigerdorf" darf höchstens 2500 Einwohner haben. mindestens ein Fünftel seiner Fläche muss als Schutzgebiet ausgewiesen sein.

Baseline: der Upper Bavarian Distrikt Ramsau und Berchtesgaden ist Deutschlands erste,, Mountaineers 'Village ". das Dorf von 1.800 Einwohnern im Berchtesgaden National Park erhielt den Preis für den,, sanften Tourismus" *von der Hand des Vizevorsitzenden* der Deutschen Alpine Association, Ludwig Wucherpfing am Mittwoch. in Österreich gibt es bereits 20,, Mounineers' Villages ". in unserem Nachbarland ist die lokale Alpine Association dafür verantwortlich, diese Unterscheidung zu vergeben. ein,, Mountagiers 'Village" darf ein Maximum von 2.500 Einwohnern haben. mindestens ein Fünftel der Gegend muss als geschütztes Gebiet ausgewiesen werden.

Ours: der Upper Bavaristische Bezirk Ramsau bei Berchtesgaden ist Deutschlands erstes "Mountaineers 'Village". das Dorf mit 1.800 Einwohnern im Berchtesgaden National Park erhielt am Mittwoch den Preis für "sanften Tourismus" *aus der Hand des Vizepräsidenten* der deutschen Alpine Association, Ludwig Wucherenning. in Österreich gibt es bereits 20 "Mountaineers' Villages". in unserem Nachbarland ist die lokale Alpine Association dafür verantwortlich, diese Unterscheidung zu vergeben. ein "Mountaineers 'Village" darf ein Maximum von 2.500 Einwohnern haben. mindestens ein Fünftel seines Gebietes muss als geschützte Gegend bezeichnet werden.

Table 9: Comparison of the document-level translations from G-Transformer (fnt.) baseline and target-side augmentation, evaluated on *News* English-German.

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
Limitations (Section 6)

□ A2. Did you discuss any potential risks of your work?
Not applicable. Left blank.

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

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

B ☑ Did you use or create scientific artifacts?

Method (Section 3): we create a new method and release the code. Experiments (Section 4): we use three benchmark datasets.

☑ B1. Did you cite the creators of artifacts you used?
Experiments (Section 4): we cite the three benchmarks.

☑ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
Introduction (Section 1): we state the public availability of our code.

☑ 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)?
Experiments (Section 4): we describe the three benchmarks briefly.

□ 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?
Not applicable. The data that we use are common benchmarks.

☑ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
Experiments (Section 4): we describe and statistics the three benchmark datasets.

☑ 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.
Experiments (Section 4): we describe and statistics the three benchmark datasets that we use.

C ☑ Did you run computational experiments?

Experiments (Section 4): we experiment with our method in various settings.

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
Training Settings (Appendix B.3)

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?
Training Settings (Appendix B.3)

☑ 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?
Training Settings in Experiments (Section 4)

☑ 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.)?
Metrics (Appendix B.2)

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.?
Not applicable. Left blank.

□ 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)?
Not applicable. Left blank.

□ 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?
Not applicable. Left blank.

□ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
Not applicable. Left blank.

□ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
Not applicable. Left blank.