

Domain Adaptive Text Style Transfer

Dianqi Li¹, Yizhe Zhang², Zhe Gan², Yu Cheng²,
Chris Brockett², Ming-Ting Sun¹, Bill Dolan²

¹University of Washington ²Microsoft Corporation

{dianqili, mts}@uw.edu

{Yizhe.Zhang, Zhe.Gan, Yu.Cheng, Chris.Brockett, billdol}@microsoft.com

Abstract

Text style transfer without parallel data has achieved some practical success. However, in the scenario where less data is available, these methods may yield poor performance. In this paper, we examine domain adaptation for text style transfer to leverage massively available data from other domains. These data may demonstrate domain shift, which impedes the benefits of utilizing such data for training. To address this challenge, we propose simple yet effective domain adaptive text style transfer models, enabling domain-adaptive information exchange. The proposed models presumably learn from the source domain to: (i) distinguish stylized information and generic content information; (ii) maximally preserve content information; and (iii) adaptively transfer the styles in a domain-aware manner. We evaluate the proposed models on two style transfer tasks (sentiment and formality) over multiple target domains where only limited non-parallel data is available. Extensive experiments demonstrate the effectiveness of the proposed model compared to the baselines.

1 Introduction

Text style transfer, which aims to edit an input sentence with the desired style while preserving style-irrelevant content, has received increasing attention in recent years. It has been applied successfully to stylized image captioning (Gan et al., 2017), personalized conversational response generation (Zhang et al., 2018a), formalized writing (Rao and Tetreault, 2018), offensive to non-offensive language transfer (dos Santos et al., 2018), and other stylized text generation tasks (Akama et al., 2017; Zhang et al., 2019).

Text style transfer has been explored as a sequence-to-sequence learning task using parallel datasets (Jhamtani et al., 2017). However, parallel datasets are often not available, and hand-

annotating sentences in different styles is expensive. The recent surge of deep generative models (Kingma and Welling, 2013; Goodfellow et al., 2014) has spurred progress in text style transfer without parallel data by learning disentanglement (Hu et al., 2017; Shen et al., 2017; Fu et al., 2018; Li et al., 2018; Prabhumoye et al., 2018). These methods typically require massive amounts of data (Subramanian et al., 2018), and may perform poorly in limited data scenarios.

A natural solution to the data-scarcity issue is to resort to massive data from other domains. However, directly leveraging abundant data from other domains is problematic due to the discrepancies in data distribution on different domains. Different domains generally manifest themselves in domain-specific lexica. For example, sentiment adjectives such as “*delicious*”, “*tasty*”, and “*disgusting*” in restaurant reviews might be out of place in movie reviews, where the sentiment words such as “*imaginative*”, “*hilarious*”, and “*dramatic*” are more typical. Domain shift (Gretton et al., 2009) is thus apt to result in feature misalignment.

In this work, we take up the problem of domain adaptation in scenarios where the target domain data is scarce and misaligned with the distribution in the source domain. Our goal is to achieve successful style transfer into the target domain, with the help of the source domain, while the transferred sentences carry relevant characteristics in the target domain.

We present two first-of-their-kind domain adaptive text style transfer models that facilitate domain-adaptive information exchange between the source and target domains. These models effectively learn generic content information and distinguish domain-specific information. Generic content information, primarily captured by modeling a large corpus from the source domain, fa-

cilitates better content preservation on the target domain. Meanwhile, domain-specific information, implicitly imposed by domain vectors and domain-specific style classifiers, underpins the transferred sentences by generating target-specific lexical terms.

Our contributions in this paper are threefold: (i) We explore a challenging domain adaptation problem for text style transfer by leveraging massively-available data from other domains. (ii) We introduce simple text style transfer models that preserve content and meanwhile translate text adaptively into target-domain-specific terms. (iii) We demonstrate through extensive experiments the robustness of these methods for style transfer tasks (sentiment and formality) on multiple target domains where only limited non-parallel data is available. Our implementation is available at <https://github.com/cookielee77/DAST>.

2 Related Work

Text Style Transfer. Text style transfer using neural networks has been widely studied in the past few years. A common paradigm is to first disentangle latent space as content and style features, and then generate stylistic sentences by tweaking the style-relevant features and passing through a decoder. Hu et al. (2017); Fu et al. (2018); Shen et al. (2017); Yang et al. (2018); Gong et al. (2019); Lin et al. (2017) explored this direction by assuming the disentanglement can be achieved in an auto-encoding procedure with a suitable style regularization, implemented by either adversarial discriminators or style classifiers. Li et al. (2018); Xu et al. (2018); Zhang et al. (2018c) achieved disentanglement by filtering the stylistic words of input sentences. Recently, Prabhumoye et al. (2018) has proposed to use back-translation for text style transfer with a de-noising auto-encoding objective (Logeswaran et al., 2018; Subramanian et al., 2018). Our work differs in that we leverage domain adaptation to deal with limited target domain data, whereas previous methods require massive target domain style-labelled samples.

Domain Adaptation. Domain adaptation has been studied in various natural language processing tasks, such as sentiment classification (Qu et al., 2019), dialogue system (Wen et al., 2016), abstractive summarization (Hua and Wang, 2017; Zhang et al., 2018b), machine translation (Koehn

and Schroeder, 2007; Axelrod et al., 2011; Sennrich et al., 2016b; Michel and Neubig, 2018), etc. However, little or no work explores domain adaptation on text style transfer. To the best of our knowledge, we are the first to explore the adaptation of text style transfer models to a new domain with limited non-parallel data available. The task requires both style transfer and domain-specific generation on the target domain. To differentiate different domains, Sennrich et al. (2016a); Chu et al. (2017) appended domain tokens to the input sentences. Our model uses learnable domain vectors combining domain-specific style classifiers, which force the model to learn distinct stylized information in each domain.

3 Preliminary

We first describe a standard text style transfer approach, which only considers data in the target domain. We limit our discussion to the scenario where only non-parallel data is available, since large amounts of parallel data is typically not feasible.

Given a set of style-labelled sentences $\mathcal{T} = \{(x_i, l_i)\}_{i=1}^N$ in the target domain, the goal is to transfer sentence x_i with style l_i to a sentence \tilde{x}_i with another style \tilde{l}_i , where $\tilde{l}_i \neq l_i$. l_i, \tilde{l}_i belong to a set of style labels $l^\mathcal{T}$ in the target domain: $l_i, \tilde{l}_i \in l^\mathcal{T}$. Typically, an encoder encodes the input x_i to a semantic representation c_i , while a decoder controls or modifies the stylistic property and decodes the sentence \tilde{x}_i based on c_i and the pre-specific style \tilde{l}_i .

Specifically, we denote an encoder-decoder model as (E, D) . The semantic representation c_i of sentence x_i is extracted by the encoder E , i.e., $c_i = E(x_i)$. The decoder D aims to learn a conditional distribution of \tilde{x}_i given the semantic representation c_i and style \tilde{l}_i :

$$p_D(\tilde{x}_i | c_i, \tilde{l}_i) = \prod_{t=1}^T p_D(\tilde{x}_i^t | \tilde{x}_i^{<t}, c_i, \tilde{l}_i), \quad (1)$$

where \tilde{x}_i^t is the t^{th} token of \tilde{x}_i , and $\tilde{x}_i^{<t}$ is the prefix of \tilde{x}_i up to the t^{th} token.

Directly estimating Eqn. (1) is impractical during training due to a lack of parallel data (x_i, \tilde{x}_i) . Alternatively, the original sentence x_i should have high probability under the conditional distribution $p_D(x_i | c_i, l_i)$. Thus, an auto-encoding reconstruc-

tion loss could be formulated as:

$$L_{ae}^{\mathcal{T}} = - \mathbb{E}_{x_i \sim \mathcal{T}} \log p_D(x_i | c_i, l_i). \quad (2)$$

Note that we assume that the decoder D recovers x_i 's original stylistic property as accurate as possible when given the style label l_i . To achieve text style transfer, the decoder manipulates the style of generated sentences by replacing l_i with a desired style \tilde{l}_i . Specifically, the generated sentence \tilde{x}_i is sampled from $\tilde{x}_i \sim p_D(\tilde{x}_i | c_i, \tilde{l}_i)$. However, by directly optimizing Eqn. (2), the encoder-decoder model tends to ignoring the style labels and collapses to a reconstruction model, which might simply copy the input sentence, hence fails to transfer the style. To force the model to learn meaningful style properties, Hu et al. (2017, 2018) apply a style classifier for the style regularization. The style classifier ensures the encoder-decoder model to transfer \tilde{x}_i with its correct style label \tilde{l}_i :

$$L_{style}^{\mathcal{T}} = - \mathbb{E}_{\tilde{x}_i \sim p_D(\tilde{x}_i | c_i, \tilde{l}_i)} \log P_{C^{\mathcal{T}}}(\tilde{l}_i | \tilde{x}_i), \quad (3)$$

where $C^{\mathcal{T}}$ is the style classifier pretrained on the target domain. The overall training objective for text style transfer within the target domain \mathcal{T} is written as:

$$L^{\mathcal{T}} = L_{ae}^{\mathcal{T}} + L_{style}^{\mathcal{T}}. \quad (4)$$

4 Domain Adaptive Text Style Transfer

In this section, we present Domain Adaptive Style Transfer (DAST) models to perform style transfer on a target domain by borrowing the strength from a source domain, while maintaining the transfer to be domain-specific.

4.1 Problem Definition

Suppose we have two sets of style-labelled sentences $\mathcal{S} = \{(x'_i, l'_i)\}_{i=1}^{N'}$, $\mathcal{T} = \{(x_i, l_i)\}_{i=1}^N$ in the source domain \mathcal{S} and the target domain \mathcal{T} , respectively. x'_i denotes the i^{th} source sentence. l'_i denotes the corresponding style label, which belongs to a source style label set: $l'_i \in l^{\mathcal{S}}$ (e.g., positive/negative). l'_i can be available or unknown. Likewise, pair (x_i, l_i) represents the sentence and style label in the target domain, where $l_i \in l^{\mathcal{T}}$.

We consider domain adaptation in two settings: (i) the source style $l^{\mathcal{S}}$ is unknown, e.g., we may have a large corpus, such as Yahoo! Answers, but the underlying style for each sample is not available; (ii) the source styles are available, and are

the same as the target styles, i.e., $l^{\mathcal{T}} = l^{\mathcal{S}}$, e.g., both IMDB movie reviews and Yelp restaurant reviews have the same style classes (negative and positive sentiments).

In both scenarios, we assume that the target domain \mathcal{T} only has limited non-parallel data. With the help of source domain data \mathcal{S} , the goal is to transfer (x_i, l_i) to $(\tilde{x}_i, \tilde{l}_i)$ in the target domain. The transferred sentence \tilde{x}_i should simultaneously hold: (i) the main content with x_i , (ii) a different style \tilde{l}_i from l_i , and (iii) domain-specific characteristics of the target data distribution \mathcal{T} .

4.2 DAST with unknown-stylized source data

In this section, we investigate the case that the source style $l^{\mathcal{S}}$ is unknown. We first examine a drawback of limited target data to motivate our method. With limited target data, Eqn. (4) may yield an undesirable transferred text, where the generated text tends to use the most discriminative words that the target style prefers while ignoring the content. This is because the classifier $C^{\mathcal{T}}$ typically requires less data to train than a sequence auto-encoder (E, D). The classifier objective $L_{style}^{\mathcal{T}}$ thus dominates Eqn. (4), rendering the generator to bias the sentences with most representative stylized (e.g., positive or negative) words rather than preserving the contents (see Table 5 for examples).

We consider alleviating this issue by leveraging massive source domain data to enhance the content-preserving ability, though the underlying styles in the source domain are unknown. By jointly training an auto-encoder on both the source and target domain data, the learned generic content information enables the model to yield better content preservation on the target domain.

To utilize the source data, we consider that $l^{\mathcal{S}}$ only contains a special unknown-style label l^u , separated from the target style $l^{\mathcal{T}}$. We assume the semantic representation of the source data c'_i is encoded by the encoder, i.e., $c'_i = E(x'_i)$. The decoder takes c'_i with style l^u to generate the sentences on the source domain. The auto-encoding reconstruction objective of the source domain is:

$$L_{ae}^{\mathcal{S}} = - \mathbb{E}_{x'_i \sim \mathcal{S}} \log p_D(x'_i | c'_i, l^u), \quad (5)$$

where the encoder-decoder model (E, D) is shared in both domains. Therefore, the corresponding objective can be written as:

$$L_{\text{DAST-C}} = L_{ae}^{\mathcal{T}} + L_{style}^{\mathcal{T}} + L_{ae}^{\mathcal{S}}. \quad (6)$$

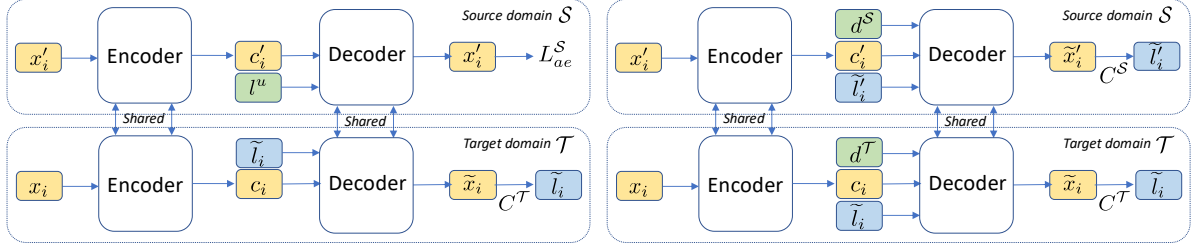


Figure 1: Illustration of the proposed **DAST-C** (left) and **DAST** (right) model. DAST-C learns the generic content information through L_{ae}^S on massive source domain data with unknown style l^u . For DAST, d^T , d^S and C^T , C^S denote domain vectors and domain-specific style classifiers, respectively. Better looked in color.

This can be perceived as combining the source domain data with the target domain data to train a better encoder-decoder framework, while target-specific style information on the target domain is learned through L_{style}^T .

Note that L_{ae}^T and L_{ae}^S are conditional on domain-specific styles labels: l^T and l^u , which implicitly encourages the model to learn domain-specific features. The decoder could thus generate target sentences adaptively with l^T , while achieving favorable content preservation with the generic content information modeled by L_{ae}^S . We refer this model, which is illustrated in Figure 1(left), as *Domain Adaptive Style Transfer with generic Content preservation (DAST-C)*.

4.3 DAST with stylized source data

We further explore the scenario where $l^S = l^T$. In this case, besides the generic content information, there is much style information from the source domain that could be leveraged, e.g., generic stylized expressions like “fantastic” and “terrible” for sentiment transfer can be applied to both restaurant and movie reviews. We thus consider to borrow the full strength of the source data, by sharing learned knowledge on both the generic content and style information.

A straightforward way to achieve this is to train Eqn. (4) on both domains. However, simply mixing the two domains together will lead to undesirable style transfers, where the transfer is not domain-specific. For example, when adapting the IMDB movie reviews to the Yelp restaurant reviews, directly sharing the style transfer model without specifying the domain will inevitably result in generations like “The pizza is dramatic!”.

To alleviate this problem, we introduce additional domain vectors, encouraging the model to perform style transfer in a domain-aware manner. The proposed DAST model is illustrated in Fig-

ure 1(right). Consider two domain vectors: d^S for the source domain and d^T for the target domain, respectively. We rewrite the auto-encoding loss as:

$$L_{ae}^{S,T} = - \mathbb{E}_{x'_i \sim S} \log p_D(x'_i | c'_i, d^S, l'_i) - \mathbb{E}_{x_i \sim T} \log p_D(x_i | c_i, d^T, l_i), \quad (7)$$

where the encoder-decoder model (E, D) is shared across domains. The domain vectors, d^S , d^T , learned from the model, implicitly guide the decoder to generate sentences with domain-specific characteristics. Note that l_i and l'_i are shared, i.e., $l^T = l^S$. This enables the model to learn generic style information from both domains. On the other hand, explicitly learning precise stylized information within each domain is crucial to generate domain-specific styles. Thus, two domain-specific style classifiers ensure the model to learn the corresponding styles by conditioning on (d^S, \tilde{l}'_i) in the source domain or (d^T, \tilde{l}_i) in the target domain:

$$L_{style}^{S,T} = - \mathbb{E}_{\tilde{x}'_i \sim p_D(\tilde{x}'_i | c'_i, d^S, \tilde{l}'_i)} \log P_{C^S}(\tilde{l}'_i | \tilde{x}'_i) - \mathbb{E}_{\tilde{x}_i \sim p_D(\tilde{x}_i | c_i, d^T, \tilde{l}_i)} \log P_{C^T}(\tilde{l}_i | \tilde{x}_i), \quad (8)$$

where $\tilde{x}'_i, \tilde{x}_i$ are the transferred sentences with pre-specific styles $\tilde{l}'_i, \tilde{l}_i$ in the source and target domains, respectively. The domain-specific style classifiers, C^T and C^S , are trained separately on each domain. The signals from classifiers encourage the model to learn domain-specific styles combining with the domain vectors and style labels. The overall training objective of the proposed DAST model is:

$$L_{DAST} = L_{ae}^{S,T} + L_{style}^{S,T}. \quad (9)$$

The domain-specific style classifiers enforce the model to learn domain-specific style information

conditioning on (d^S, \tilde{l}_i') or (d^T, \tilde{l}_i) , which in turn controls the model to generate sentences with domain-specific words. The model can thus distinguish domain-specific features, and adaptively transfer the styles in a domain-aware manner.

5 Experiments

We evaluate our proposed models on two tasks: sentiment transfer (positive-to-negative and negative-to-positive), and formality transfer (informal-to-formal). In both tasks, we make comparisons with previous approaches over multiple target domains. All experiments are conducted on one Nvidia GTX 1080Ti GPU.

5.1 Dataset

A statistics for the source and target corpora used in the experiments is summarized in Table 1.

Sentiment Transfer					
Source	Train	Target	Train	Dev	Test
IMDB	344k	YELP	444k	4k	1k
		AMAZON	554k	2k	1k
		YAHOO	4k	2k	1k
Formality Transfer					
Source	Train	Target	Train	Dev	Test
GYAFC	103k	ENRON	6k	500	500

Table 1: Statistics of source and target datasets.

Sentiment Transfer. For the source domain, we use IMDB movie review corpus (Diao et al., 2014) by following the filtering and preprocessing pipelines from Shen et al. (2017). This results in 344k training samples with sentiment labels. For the target domain, both the Yelp restaurant review dataset and the Amazon product review dataset are from Li et al. (2018). For the test sets, we evaluate our methods by using 1k human-transferred sentences, annotated by Li et al. (2018), on both Yelp and Amazon datasets. In addition to the two standard sentiment datasets, we manually collected a Yahoo sentimental question dataset - 7k question samples with sentiments from Yahoo! Answers dataset (Zhang et al., 2015). We split the 7k sentimental questions into 4k/2k/1k for train/dev/test sets, respectively. Note that the Yahoo sentiment dataset only consists of questions, which have different domain characteristics with the IMDB dataset. In all the sentiment experiments, we consider both transfer directions (positive-to-negative and negative-to-positive).

Formality Transfer. We use Grammarly’s Yahoo Answers Formality Corpus (GYAFC) (Rao

and Tetreault, 2018) as the source dataset. The publicly released version of GYAFC only covers two topics (*Entertainment & Music* and *Family & Relationships*), where each topic contains 50k paired informal and formal sentences written by humans. For the target domain, we use Enron email conversation dataset¹, which covers several different fields like *Business*, *Politics*, *Daily Life*, etc. We manually labeled 7k non-parallel sentences written in either the formal or informal style. We split the Enron dataset into 6k, 500, 500 samples for training, validation and testing, respectively. Both the validation and test set consist of mere informal sentences, where the corresponding formal references are annotated by us from a crowd-sourcing platform for evaluation. We only assess the informal-to-formal transfer direction in the formality transfer experiment.

5.2 Evaluation

Automatic Metrics. We evaluate the effectiveness of our DAST models based on three automatic metrics:

(i) **Content Preservation.** We assess the content preservation according to n-gram statistics, by measuring the BLEU scores (Papineni et al., 2002) between generated sentences and human references on the target domain, referred as *human BLEU* (*hBLEU*). When no human reference is available (e.g., Yahoo), we compute the BLEU scores with respect to the input sentences.

(ii) **Style Control.** We generate samples from the model and measure the style accuracy with a style classifier that is pre-trained on the target domain. We refer the style accuracy as S-acc.

(iii) **Domain Control.** To validate whether the generated sentences hold the characteristics of the target domain, we adopt a pre-trained domain classifier to measure the percentage of generated sentences that belong to the target domain. We refer the domain accuracy as D-acc.

All the pre-trained classifiers are implemented by TextCNN (Kim, 2014; Zhang et al., 2017). The test accuracy of all these classifiers used for evaluation are reported in Appendix A.1. Following Xu et al. (2018), we also evaluate all methods using a single unified metric called G-score, which calculates the geometric mean of style accuracy and *hBLEU*.

¹<https://www.cs.cmu.edu/~Enron/>

Model (100% target data)	Yelp				Amazon			
	D-acc	S-acc	<i>h</i> BLEU	G-score	D-acc	S-acc	<i>h</i> BLEU	G-score
CrossAlign (Shen et al., 2017)	-	85.0	3.7	8.3	-	23.0	34.1	18.0
Delete&Retrieve (Li et al., 2018)	-	90.6	14.8	17.9	-	50.9	30.3	25.7
CycleRL (Xu et al., 2018)	-	88.7	12.3	16.4	-	68.7	14.2	15.5
SMAE (Zhang et al., 2018c)	-	85.1	12.1	15.5	-	71.1	12.9	14.9
ControlGen (Hu et al., 2018)	-	91.5	25.5	27.4	-	79.0	31.1	30.5
Finetune	96.1	91.3	25.6	27.8	97.4	79.2	34.1	34.3
DAST-C (ours)	93.8	91.7	25.7	27.5	96.7	81.9	35.7	35.0
DAST (ours)	95.8	92.3	26.3	28.9	96.9	83.0	35.9	35.1
Model (1% target data)	D-acc	S-acc	<i>h</i> BLEU	G-score	D-acc	S-acc	<i>h</i> BLEU	G-score
CrossAlign (Shen et al., 2017)	-	76.3	4.8	8.5	-	83.2	2.0	5.9
Delete&Retrieve (Li et al., 2018)	-	82.1	4.1	7.6	-	63.0	6.9	9.3
CycleRL (Xu et al., 2018)	-	86.6	1.4	5.2	-	79.5	0.7	3.8
SMAE (Zhang et al., 2018c)	-	96.0	1.2	4.8	-	87.2	0.4	3.2
ControlGen (Hu et al., 2018)	-	98.5	3.7	8.6	-	83.2	1.9	5.8
Finetune	98.1	96.7	13.9	18.5	96.0	89.2	11.3	14.4
DAST-C (ours)	96.9	90.3	17.8	19.3	94.8	78.2	20.1	21.6
DAST (ours)	97.0	92.6	20.1	23.1	94.6	82.7	21.0	23.1

Table 2: Automatic evaluation results on Yelp and Amazon test sets. D-acc and S-acc denote domain accuracy and style accuracy, respectively. G-score is the geometric mean of S-acc and *h*BLEU.

Human Evaluation. To assess the quality of transferred sentences, we conduct human evaluations based on the facets of *content preservation*, *style control* and *fluency*, following Mir et al. (2019). Previous works (Subramanian et al., 2018; Gong et al., 2019) ask workers to evaluate the quality via a numerical score, however, we found that this empirically leads to high-variance results. Instead, we pair transferred sentences from two different models, and ask workers to choose the sentence they prefer *when compared to the input* on each evaluation aspect. We provide a “No Preference” option to choose when the workers think the qualities of the two sentences are indistinguishable. Details of the human evaluation instruction are included in Appendix A.3. For each testing, we randomly sample 100 sentences from the corresponding test set and collect three human responses for each pair on every evaluation aspect, resulting in 2700 responses in total.

5.3 Experimental Setup

The encoder E and the decoder D are implemented by one-layer GRU (Cho et al., 2014) with hidden dimensions 500 and 700, respectively. The domain-vector dimension is set to 50. The style labels are represented by learnable vectors with 150 dimensions. The decoder is initialized by a concatenation of representations of content, style, and domain vectors. If domain vectors are not used, the dimension of style labels is set to 200;

accordingly, the initialization of the decoder is a concatenation of content and style representations. TextCNN (Kim, 2014) is employed for the domain-specific style classifiers pre-trained on corresponding domains. After pre-training, the parameters of the classifiers are fixed. We use the hard-sampling trick (Logeswaran et al., 2018) to back-propagate the loss through discrete tokens from the classifier to the encoder-decoder model. During training, we assign each mini-batch the same amount of source and target data to balance the training.

We make an extensive comparison with five state-of-the-art text style transfer models: CrossAlign (Shen et al., 2017), Delete&Retrieve (Li et al., 2018), CycleRL (Xu et al., 2018), SMAE (Zhang et al., 2018c) and ControlGen (Hu et al., 2018). We also experiment a simple and effective domain adaptation baseline - Finetune, which is trained with Eqn. (4) on the source domain and then fine-tuned on the target domain.

5.4 Results

Model Comparisons. To evaluate the effectiveness of leveraging massive data from other domains, we compare our proposed DAST models with previously proposed models trained on the target domain (Table 2). We observe that by leveraging massive data from the IMDB dataset, our models achieve better performance against all

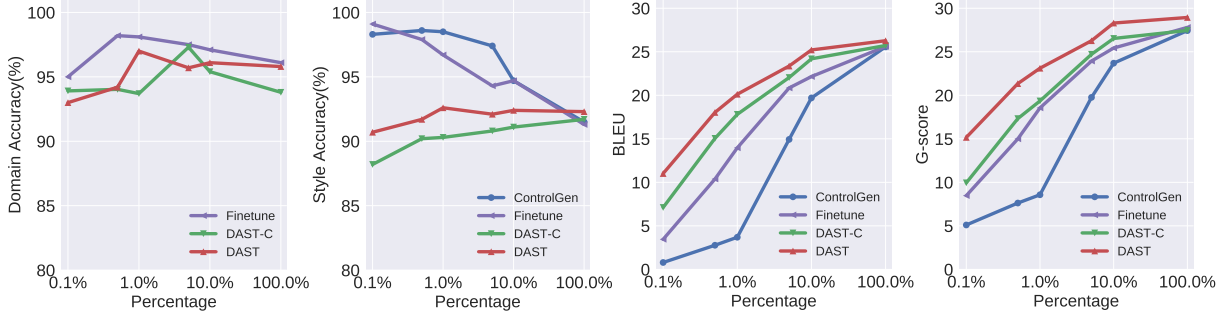


Figure 2: Results on Yelp test set in terms of different percentage of target domain data. 0.1% \approx 400 samples.

Style Control (Yelp 1% data)				Content Preservation (Yelp 1% data)				Fluency (Yelp 1% data)						
Our Model	Neutral	Comparison		Our Model	Neutral	Comparison		Our Model	Neutral	Comparison				
DAST	56.2%	30.5%	13.3%	ControlGen	DAST	47.0%	48.4%	4.6%	ControlGen	DAST	47.1%	40.8%	12.0%	ControlGen
DAST	40.5%	42.3%	17.2%	DAST-C	DAST	22.4%	65.7%	11.9%	DAST-C	DAST	29.1%	55.8%	15.1%	DAST-C
DAST	17.9%	18.5%	63.6%	Human	DAST	17.7%	47.4%	34.9%	Human	DAST	10.1%	30.4%	59.5%	Human

Style Control (Enron)				Content Preservation (Enron)				Fluency (Enron)						
Our Model	Neutral	Comparison		Our Model	Neutral	Comparison		Our Model	Neutral	Comparison				
DAST	74.2%	19.8%	6%	ControlGen	DAST	80.8%	14.8%	4.4%	ControlGen	DAST	73.8%	20.6%	5.6%	ControlGen
DAST	28.4%	50.2%	21.4%	DAST-C	DAST	26.8%	48.8%	24.4%	DAST-C	DAST	26.9%	51.6%	21.5%	DAST-C
DAST	17.6%	30.5%	51.9%	Human	DAST	15.3%	36.9%	47.8%	Human	DAST	11.6%	36.5%	51.9%	Human

Table 3: Results of **Human Evaluation** for style control, content preservation and fluency, showing preferences (%) for DAST model vis-a-vis baseline or other comparison systems. Evaluation results of the **overall transfer quality** are provided in Appendix A.3.

Yahoo Sentiment Transfer			
Model	D-acc	S-acc	BLEU
ControlGen	-	99.1	9.7
Finetune	97.8	98.8	31.4
DAST-C	90.7	98.8	35.9
DAST	90.8	99.2	39.2

Table 4: Results on Yahoo sentiment transfer task.

baselines on the sentiment transfer tasks in both the Yelp and Amazon domains.

Notably, when the target domain has limited data (1%), all baselines trained only on the target domain fail completely on content preservation. Finetune preserves better content but experiences the catastrophic forgetting problem (Goodfellow et al., 2013) to the source domain information. As a result, the overall style transfer performance is still nonoptimal. By contrast, with the help of the source domain, DAST obtains considerable content preservation performance improvement when compared with other baselines. Our model also attains favorable performance in terms of style transferring accuracy (S-acc), resulting in a good overall G-score. In general, we observe that DAST-C is able to better preserve content information, while DAST further improves both content preservation and style control abilities. Additionally,

both DAST-C and DAST can adapt to the target domain, as evidenced by the high domain accuracy (D-acc). The human evaluation results (Table 3) show a strong preference of DAST over DAST-C as well as ControlGen in terms of style control, content preservation and fluency.

Finally, we evaluate our models on Yahoo sentiment transfer task. As can be seen in Table 4, both DAST and DAST-C achieve successful style transfer even if the target data is formed as questions which have a large discrepancy with the source IMDB domain. The samples of Yelp and Yahoo sentiment transfer are shown in Table 5. We also investigate the effect of different source domain data, included in Appendix A.2.

Limiting the Target Domain Data. We further test the limit of our model by using as few target domain data as possible. Figure 2 shows the quantitative results with different percentages of target domain training data. When the target domain data is insufficient, especially less than 10%, the content preservation ability of the baseline (trained with target data only) has degenerated rapidly despite a relatively high style transfer accuracy. This is less than desirable because by retrieving sentences with the target style a transferred sentence

	Yelp (positive-to-negative)	Yelp (negative-to-positive)
Input	the service was great , food delicious , and the value impeccable .	and the pizza was cold , greasy , and generally quite awful .
ControlGen	the service was horrible , service , the service and very frustrated .	and the food was delicious , delicious , and freaking tasty , delicious .
Finetune	the service was poor , food ... , and the experience were .	and the pizza was professional , friendly , and always have great .
DAST-C	the service was horrible , food horrible , and the slow sparse .	and the pizza was fresh , greasy , and generally quite cool .
DAST	the service was horrible , food bland , and the value lousy .	and the pizza was tasty , juicy , and definitely quite amazing .
Human	service was poor and the food expensive and weak tasting .	the pizza was warm , not greasy , and generally tasted great .
	Yahoo (positive-to-negative)	Yahoo (negative-to-positive)
Input	who is more romantic ? man or woman ?	why do stupid questions constantly receive intelligent answers ?
ControlGen	which is more stupid ? and or why ?	men do fantastic questions constantly receive intelligent bound !
Finetune	the is more expensive ? man or woman ?	why do great questions read more entertaining answers ?
DAST-C	who is more ugly ? man or woman ?	why do important questions constantly receive intelligent answers ?
DAST	who is more crazy ? man or woman ?	why do nice questions constantly receive intelligent answers ?
	Enron (informal-to-formal)	Enron (informal-to-formal)
Input	ya 'll need to come visit us in austin .	are n't you suppose to be teaching some kids or something ?
ControlGen	could we need to look on saturday in enpower .	are you not supposed to be disloyal some kids or something ?
Finetune	you will need to go in bed with him .	are you not to be able to be some man or something ?
DAST-C	you will need to visit town .	are not you supposed to be teaching some kids or something ?
DAST	yes , you will need to visit us in austin .	are you not supposed to be teaching some children or something ?
Human	all of you should come visit us in austin .	are you not supposed to be instructing children ?

Table 5: Transferred sentences on Yelp (1% data), Yahoo and Enron datasets, where **red** denotes successful style transfers, **blue** denotes content losses, and **orange** denotes grammar errors. Better looked in color.

Model	D-acc	S-acc	<i>h</i> BLEU	G-score
DAST	97.0	92.6	20.1	23.1
w/o d-spec attributes	83.9	90.9	20.0	22.7
w/o d-spec classifiers	91.4	83.8	19.0	20.8
w/o both	73.8	80.6	18.7	19.9
Setup	D-acc	S-acc	<i>h</i> BLEU	G-score
IMDB+Yelp	97.0	92.6	20.1	23.1
Finetune	98.1	96.7	13.9	18.5
IMDB	62.8	59.3	21.4	12.2
Yelp	96.8	98.5	3.7	8.6

Table 6: Ablation study on Yelp (1%) dataset with help from IMDB dataset. The results are evaluated on Yelp test set. *d-spec* is short for *domain-specific*.

can easily exhibit the correct style while retaining barely any content similar to the input. Finetune improves content preservation but still suffers the same problem with less target data. Note that DAST-C is not comparable to Finetune as the former does not use the style information in the source domain.

Both DAST models bring substantial improvements to content preservation, and can still successfully manipulate the styles, resulting in consistently higher G-scores. This is presumably because our models adapt the content information as well as the style information from the source domain to consistently sustain the style transfer on the target domain. By learning both generic and domain-specific stylized information, DAST outperforms DAST-C in terms of content preservation and style control. Even with 0.1% target domain data (400 samples), DAST was able to attain a reasonable degree of text style transfer, whereas the model trained on the target data generated entirely nonsensical sentences. Meanwhile, DAST succeeded in transferring the sentences in a domain-

Enron Formality Transfer			
Model	D-acc	S-acc	<i>h</i> BLEU
ControlGen	-	81.2	4.74
Finetune	91.3	81.6	14.7
DAST-C	87.6	89.2	15.5
DAST	88.4	91.6	16.4

Table 7: Results on Enron formality transfer tasks.

aware manner, achieving consistently high domain accuracy.

Ablation Study. To investigate the effect of individual components and training setup on the overall performance, we conduct an ablation study in Table 6. The domain vectors enable the model to transfer sentences in a domain-aware manner, and thus give the largest boost on domain accuracy. Without domain-specific style classifiers, the model mixes the style information on both domains, resulting in worse style control and content preservation. Additionally, simply increasing the number of training examples (i.e., the row “w/o both”) improves content preserving, while introducing a data distribution discrepancy between the training (Yelp+IMDB) and test data (Yelp), as evidenced by the lower S-acc and D-acc scores.

In terms of the training setup, the source domain IMDB mostly helps content preservation, while accurate style information is mainly learned from the target domain Yelp. Finetune gives higher S-acc and D-acc and lower *h*BLEU due to catastrophic forgetting. Our proposed DAST successfully exploits the source domain data, and thus yields balanced results on style and domain control, while maintaining content preservation.

Non-parallel Style Transfer with Parallel Source Data. Finally, to verify the versatility of our proposed models in different scenarios, we investigate another domain adaptation setting, where the source domain data (GYAFC) is parallel but the target domain data (Enron) is non-parallel. Since parallel data is available in the source domain, we are able to simply add a sequence-to-sequence loss L_{s2s}^S on source domain data in Eqn. (6) and Eqn. (9) to help the target domain without parallel data. The training objectives can be written as: $L_{ae}^T + L_{style}^T + L_{ae}^S + L_{s2s}^S$ and $L_{ae}^{S,T} + L_{style}^{S,T} + L_{s2s}^S$, respectively. Results are summarized in Table 7. DAST outperforms other methods on both style control and content preservation while keeping the transferred sentences with target-specific characteristics (D-acc). A strong human preference for DAST can be observed in Table 3 when compared to the baselines. Qualitative samples are provided in Table 5.

6 Conclusion

We present two simple yet effective domain adaptive text style transfer models that leverage massively available data from other domains to facilitate the transfer task in the target domain. The proposed models achieve better content preservation with the generic information learned from the source domain and simultaneously distinguish the domain-specific information, which enables the models to transfer text in a domain-adaptive manner. Extensive experiments demonstrate the robustness and applicability on various scenarios where the target data is limited.

Acknowledgments

We would like to thank the reviewers for their constructive comments. We thank NVIDIA Corporation for the donation of the GPU used for this research. We also thank Hao Peng and Tianyi Zhou for their helpful discussions.

References

Reina Akama, Kazuaki Inada, Naoya Inoue, Sosuke Kobayashi, and Kentaro Inui. 2017. Generating stylistically consistent dialog responses with transfer learning. In *IJCNLP*.

Amittai Axelrod, Xiaodong He, and Jianfeng Gao. 2011. Domain adaptation via pseudo in-domain data selection. In *EMNLP*.

Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using rnn encoder-decoder for statistical machine translation. In *EMNLP*.

Chenhui Chu, Raj Dabre, and Sadao Kurohashi. 2017. An empirical comparison of simple domain adaptation methods for neural machine translation. *arXiv preprint arXiv:1701.03214*.

Qiming Diao, Minghui Qiu, Chao-Yuan Wu, Alexander J Smola, Jing Jiang, and Chong Wang. 2014. Jointly modeling aspects, ratings and sentiments for movie recommendation (jmars). In *SIGKDD*.

Zhenxin Fu, Xiaoye Tan, Nanyun Peng, Dongyan Zhao, and Rui Yan. 2018. Style transfer in text: Exploration and evaluation. In *AAAI*.

Chuang Gan, Zhe Gan, Xiaodong He, Jianfeng Gao, and Li Deng. 2017. Stylenet: Generating attractive visual captions with styles. In *CVPR*.

Hongyu Gong, Suma Bhat, Lingfei Wu, Jinjun Xiong, and Wen-mei Hwu. 2019. Reinforcement learning based text style transfer without parallel training corpus. *arXiv preprint arXiv:1903.10671*.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. In *NeurIPS*.

Ian J Goodfellow, Mehdi Mirza, Da Xiao, Aaron Courville, and Yoshua Bengio. 2013. An empirical investigation of catastrophic forgetting in gradient-based neural networks. *arXiv preprint arXiv:1312.6211*.

Arthur Gretton, AJ. Smola, Jiayuan Huang, Marcel Schmittfull, Karsten Borgwardt, and Bernhard Schölkopf. 2009. *Covariate shift and local learning by distribution matching*. MIT Press.

Zhiting Hu, Haoran Shi, Zichao Yang, Bowen Tan, Tiancheng Zhao, Junxian He, Wentao Wang, Lianhui Qin, Di Wang, et al. 2018. Texar: A modularized, versatile, and extensible toolkit for text generation. *arXiv preprint arXiv:1809.00794*.

Zhiting Hu, Zichao Yang, Xiaodan Liang, Ruslan Salakhutdinov, and Eric P Xing. 2017. Toward controlled generation of text. In *ICML*.

Xinyu Hua and Lu Wang. 2017. A pilot study of domain adaptation effect for neural abstractive summarization. In *Proceedings of the Workshop on New Frontiers in Summarization*.

Harsh Jhamtani, Varun Gangal, Eduard Hovy, and Eric Nyberg. 2017. Shakespearizing modern language using copy-enriched sequence to sequence models. In *Proceedings of the Workshop on Stylistic Variation*.

- Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *EMNLP*.
- Diederik P Kingma and Max Welling. 2013. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.
- Philipp Koehn and Josh Schroeder. 2007. Experiments in domain adaptation for statistical machine translation. In *Proceedings of the second workshop on statistical machine translation*.
- Juncen Li, Robin Jia, He He, and Percy Liang. 2018. Delete, retrieve, generate: a simple approach to sentiment and style transfer. In *NAACL*.
- Kevin Lin, Dianqi Li, Xiaodong He, Zhengyou Zhang, and Ming-Ting Sun. 2017. Adversarial ranking for language generation. In *NeurIPS*.
- Lajanugen Logeswaran, Honglak Lee, and Samy Bengio. 2018. Content preserving text generation with attribute controls. In *NeurIPS*.
- Paul Michel and Graham Neubig. 2018. Extreme adaptation for personalized neural machine translation. In *ACL*.
- Remi Mir, Bjarke Felbo, Nick Obradovich, and Iyad Rahwan. 2019. Evaluating style transfer for text. *arXiv preprint arXiv:1904.02295*.
- Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *ACL*.
- Shrimai Prabhumoye, Yulia Tsvetkov, Ruslan Salakhutdinov, and Alan W Black. 2018. Style transfer through back-translation. In *ACL*.
- Xiaoye Qu, Zhikang Zou, Yu Cheng, Yang Yang, and Pan Zhou. 2019. Adversarial category alignment network for cross-domain sentiment classification. In *NAACL*.
- Sudha Rao and Joel Tetreault. 2018. Dear sir or madam, may i introduce the gyafc dataset: Corpus, benchmarks and metrics for formality style transfer. In *NAACL*.
- Cicero Nogueira dos Santos, Igor Melnyk, and Inkit Padhi. 2018. Fighting offensive language on social media with unsupervised text style transfer. In *ACL*.
- Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016a. Controlling politeness in neural machine translation via side constraints. In *NAACL*.
- Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016b. Improving neural machine translation models with monolingual data. In *ACL*.
- Tianxiao Shen, Tao Lei, Regina Barzilay, and Tommi Jaakkola. 2017. Style transfer from non-parallel text by cross-alignment. In *NeurIPS*.
- Sandeep Subramanian, Guillaume Lample, Eric Michael Smith, Ludovic Denoyer, Marc’Aurelio Ranzato, and Y-Lan Boureau. 2018. Multiple-attribute text style transfer. *arXiv preprint arXiv:1811.00552*.
- Tsung-Hsien Wen, Milica Gašić, Nikola Mrkšić, Lina M Rojas-Barahona, Pei-Hao Su, David Vandyke, and Steve Young. 2016. Multi-domain neural network language generation for spoken dialogue systems. In *NAACL*.
- Jingjing Xu, Sun Xu, Qi Zeng, Xiaodong Zhang, Xuancheng Ren, Houfeng Wang, and Wenjie Li. 2018. Unpaired sentiment-to-sentiment translation: A cycled reinforcement learning approach. In *ACL*.
- Zichao Yang, Zhiting Hu, Chris Dyer, Eric P Xing, and Taylor Berg-Kirkpatrick. 2018. Unsupervised text style transfer using language models as discriminators. In *NeurIPS*.
- Saizheng Zhang, Emily Dinan, Jack Urbanek, Arthur Szlam, Douwe Kiela, and Jason Weston. 2018a. Personalizing dialogue agents: I have a dog, do you have pets too? In *ACL*.
- Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. In *NeurIPS*.
- Ye Zhang, Nan Ding, and Radu Soricut. 2018b. Shaped: Shared-private encoder-decoder for text style adaptation. In *NAACL*.
- Yi Zhang, Jingjing Xu, Pengcheng Yang, and Xu Sun. 2018c. Learning sentiment memories for sentiment modification without parallel data. In *EMNLP*.
- Yizhe Zhang, Xiang Gao, Sungjin Lee, Chris Brockett, Michel Galley, Jianfeng Gao, and Bill Dolan. 2019. Consistent dialogue generation with self-supervised feature learning. *arXiv preprint arXiv:1903.05759*.
- Yizhe Zhang, Dinghan Shen, Guoyin Wang, Zhe Gan, Ricardo Henao, and Lawrence Carin. 2017. Deconvolutional paragraph representation learning. In *NeurIPS*.