Applied Intelligence (2021) 51:4609–4622
https://doi.org/10.1007/s10489-020-02077-5

# Non-parallel text style transfer with domain adaptation and an attention model

**Mingxuan Hu**[1] · **Min He**[1]

Accepted: 9 November 2020 / Published online: 5 January 2021


**Abstract**

Text style transfer, the aim of which is to convert a specific style in a given sentence to another target style while maintaining the style-independent content information of the original sentence, can face challenges when applied to non-parallel text. In this paper, we combine domain adaptation learning and an attention model to propose a new framework to accomplish the task. Domain adaptation can leverage relative information from the source domain to improve the generative model's capacity for reconstructing data. The attention model can give the importance weights of generated words for the target style in a sentence; therefore, the generative model can concentrate on generating words with higher importance weights to accomplish text style transfer effectively. We evaluate our framework using Yelp, Amazon and Captions corpora. The results of automatic and human evaluation demonstrate the effectiveness of our framework compared with previous works under non-parallel and limited training data. The available codes are in https://github.com/mingxuan007/text-style-transfer-with-adversarial-network-and-domain-adaptation.

**Keywords** Text style transfer · Non-parallel text · Domain adaptation · Attention model

## 1 Introduction

One sentence can contain multiple style labels from different perspectives (e.g., "*the food makes me feel like I'm sleeping in the grass*" can simultaneously embody "*positive*" and "*romantic*" styles), and under given transfer demands for different styles in sentences, the partition between style and content can differ. Therefore, the aim of text style transfer is to change a specific style of a sentence to the desired style while maintaining the original style-independent content for a given context. For example, consider unfavorable reviews of events and merchandise with a "*negative*" style to be modified to convey a "*positive*" style through sentiment modification (one kind of text style transfer). Alternatively, one might need to modify the sentence "*the question is hard; you need to teach me*" to become a sentence with "*formal*" style such as "*the question is difficult for me; can you teach me?*" when presented to other people. In the era of big data, considerable amounts of text need to be converted to create desirable styles for different demands and audiences, and doing so demands considerable effort from editorial staff. Hence, text style transfer technology has become a hot research topic in recent years.

Due to the substantial progress made in deep learning, some text generation tasks have achieved considerable success using parallel data [29, 31]. However, as [20, 27, 32] note, when one only has access to limited non-parallel data, in which sentence pairs with the same content but specific opposite styles are difficult to find, it is difficult for conventional deep learning models to distinguish the specific style and the style-independent content of a given sentence.

To solve this problem, many studies have considered various methods and achieved different degrees of breakthrough. Generally, for non-parallel text style transfer, one first obtains style-independent content representations of the original sentences with a designed encoder, and then the decoder utilizes the representations to generate transferred sentences with the target style information. As a result of the original constraint regarding the amount of style information and content information in the original sentence,

✉ Min He
hemin@ynu.edu.cn

Mingxuan Hu
djspgo@163.com

1 School of Information Science and Engineering, Yunnan University, Kunming 650091, China

to obtain style-independent representations, [27] design the cross-aligned encoder that contains adversarial networks for discriminating styles to obtain style-independent representations with the input of the original sentences and their given style labels. However, the vanishing gradient of conventional adversarial networks cannot guarantee the quality of the training [1]. [22] delete relative style markers (n-grams), which often occur in sentences to mark the corresponding style labels in every sentence, then feed the remaining "*content*" words and extra noise into the encoder to generate style-independent representations. Since the token distributions of different sentences in a given corpus can be different, evaluating the style properties of every sentence using the marker frequency statistic of overall samples yields inaccurate results.

Another problem is the nondifferentiability of discrete token words in output sentences, so we cannot directly optimize the generator (encoder and decoder) to ensure transferred sentences with target style properties through back-propagation, such as with image data [5, 8, 36]. To address this problem, [21, 32, 35] adopt REINFORCE [28] to encourage the relative model learning policy to obtain the maximum return of the designed reward mechanism. As a result of the designed difficulty of the reward mechanism and the high variance of the sampled gradient, REINFORCE is often unstable. Some works [27] apply professor forcing [19] to match the hidden representations of discrete outputs and then utilize the representations, which have smooth distributions, to accomplish back-propagation instead of the outputs.

In this paper, we adopt an adversarial network to obtain the style-independent information representation of one sentence. Concretely, we utilize an autoencoder with a sentence to preliminarily generate the content representation (vector) and apply reconstruction training to make the representation contain enough information of the original sentence; we then design a discriminator to deliberately discriminate the specific style information from the content representation, meanwhile utilizing the encoder to reduce the style information in the representation relying on an adversarial manner with the discriminator. To avoid the problem of the nondifferentiability of transferred sentences, we apply Gumbel-Softmax [12, 18] to obtain the approximate and continuous representation of the transferred sentence. Then, to guarantee high style transfer accuracy, we utilize an attention model [2] as the style classifier to determine the target style loss for the continuous representation and back-propagate the loss to the generator (encoder and decoder), thereby minimizing it to append the target style properties in transferred sentences. The attention model can give the importance weights of generated words for the target style in a sentence, which allows the generator to concentrate on generating words with the higher importance weights in transferred sentences, thus allowing the framework accomplish text style transfer in an effective and interpretable way.

However, latent representations of original sentences always lose some information contained in the original sentences when training data are limited. To improve the encoder-decoder framework's capacity to capture sentence information, we adopt domain adaptation for the reconstruction of original sentences. To explore the useful features for the target domain and reduce domain discrepancy, we only preserve the sentences, which consist of the words of the target domain, in the source domain. In the experiment, domain adaptation greatly enhances the performance of the whole framework in the context of limited training data.

In summary, the major advantages of our work can be described in four points:

- We utilize an autoencoder to produce the style-independent content representation of one sentence preliminarily; to exclude the irrelevant style information from the representation, we train a discriminator to deliberately discriminate the special style information from it and adjust the encoder to reduce the style information in it, thereby allowing the content vectors to exclude irrelevant style information and carry more original content information.
- In the case of limited non-parallel text, we employ domain adaptation in the reconstruction to utilize similar features from the source domain by choosing the sentences that only contain the words of the target domain.
- We adopt an attention model as the style classifier to give the importance weights of generated words for the target style, which helps the generator (encoder and decoder) append target style properties in transferred sentences in an effective and interpretable fashion.
- We design different style transfer experiments to explore the practicability and efficiency of our framework. Automatic evaluation and human evaluation are adopted to evaluate the transferred sentences. Finally, we perform an ablation test on one corpus with limited training data to demonstrate the contributions of some important objectives of our framework.

## 2 Related work

Recently, research on style transfer has undergone substantial development [8, 23, 34, 36]. In the computer vision literature, [8] integrate the content features of one image, which is extracted by VGG19, with the style features of another image to generate a new image; [36] adopts a cycle-consistent adversarial network to preserve the content features of the original images for high-quality generated images. There are also many works on non-parallel text style transfer, and in this section, we introduce some of the studies related to our work.

**Reinforcement learning** [32] design a self-attention-based neutralization model to obtain the style-independent content words of sentences and two kind of decoders for reconstructing and transferring sentences through appending positive and negative sentiment information in these content words respectively (since the work focuses on sentiment modification – a type of text style transfer). They regard the training of the whole framework as reinforcement learning and utilize policy gradients to optimize these discrete outputs while obtaining more expected rewards. However, the training performance often suffers from the high variance of sampled gradients, leading to limited effectiveness.

**Disentangled representation learning** [13] first use disentangled representation learning [3] to obtain style representation vectors and content representation vectors of sentences, then combine the content representation vectors with target style vectors to generate transferred sentences through one decoder. In the disentangled representation learning, they utilize the bag-of-word (BOW) features of sentences, which exclude specific style words and stopwords, to approximate the content information and adopt multitask learning to align the content vectors of sentences with the BOW features for original content preservation. However, excluding the style words of one sentence, which depends on the word frequency statistics of the entire sentence set about style labels, is not accurate, since the feature distributions of different sentences can be different.

**Toward controlled generation of text** [11] attempt to learn the embedding representations of style labels to control the generation of specific style properties in transferred sentences by one decoder. Specifically, the decoder is optimized by learning recovering target style vectors with a style discriminator and the content vectors of original sentences with one encoder (which can be regarded as a content discriminator), which greatly promotes the quality of transferred sentences. However, compared with our work, they do not add a specific constraint for filtering style information in the content vectors, which lowers the holistic performance of their models.

**Domain adaptation** There are also many cases of domain adaptation in natural language processing, e.g., sequence tagging [25], sentiment analysis [9], and dialogue systems [30]. As the first attempt to implement domain adaptation for text style transfer, [20] append domain information in the latent representations of sentences to help the designed generative model generate domain-specific style properties in transferred sentences. However, this work focuses on the learning of style information with the source domain but ignores the preservation of original style-independent content information.

## 3 Our methods

Assuming that $X = \{x_i\}_{i=1}^{n}$ is the sentence set from target domain $\mathcal{T}$ in which each sentence is labeled with $s1$ or $s2$, $s1$ and $s2$ are contained in all possible styles of $\mathcal{T}$. $\forall x \in X$, our task is to convert a specific style $s_x^o$ in $x$ to another style $s_x^t$ while holding the style-independent content information unchanged, and $s_x^o, s_x^t \in \{s1, s2\}$. Since $X$ is a non-parallel corpus, it is difficult to find sentence pairs – $(x_{i1}, x_{i2})$ that describe the same style-independent content using different styles. The processing overview is given in Fig. 1.

### 3.1 Domain adaptation for reconstruction

Compared with the variational autoencoder (VAE) [16], which applies KL divergence [17] to constrain the encoder, we choose an autoencoder – the deterministic autoencoder (DAE) – as our generative model since it can ensure that the latent representations contain as much of the information from the original sentences as possible, rather than restraining latent representations following a simple distribution [27]. In this paper, the encoder and decoder in the DAE adopt a one-layer gated recurrent unit [4] (GRU). The content representation $z_x$, which should contain sufficient content information from the original sentence $x$, is generated preliminarily by $z_x = Encoder(x)$, where $Encoder$ represents the encoding process of GRU. Then, the decoder generates the target sentence $x' = Decoder(z_x; s_x^o)$ by learning the conditional distribution with $s_x^o$:

$$p_G(x'|z_x, s_x^o) = \prod_i p(w_i|z_x, s_x^o, w_1, w_2, ..., w_{i-1}) \tag{1}$$

where $w_i$ represents the $i^{th}$ token (word) in $x'$.

To improve the autoencoders' capacities for encoding data and allow $z_x$ to carry more content information, the reconstruction loss of $x$ is built, and $x'$ should be set equal to $x$ when minimizing the reconstruction loss. The reconstruction loss is given by:

$$\mathcal{L}_{rec}(\theta_E, \theta_G, W_s, b_s; x) = -E_{x \sim X}[\log p_G(x|z_x, s_x^o)] \tag{2}$$

where $\theta_E, \theta_G$ represent the parameters of the encoder and the decoder, respectively; $W_s$ and $b$ are the trainable weight and bias matrices to build the style embeddings of $s_x^o$ for the decoder, respectively. Concretely, we let the style embeddings of $s1$ and $s2$ be $1 * W_s + b$ and $0 * W_s + b$, respectively, then concatenate the style embedding, which

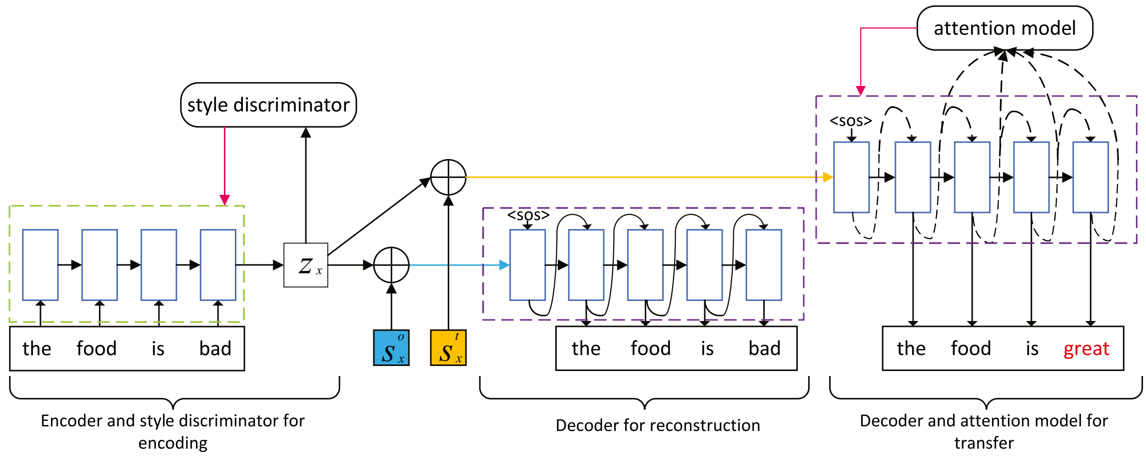



**Fig. 1** The processing overview assuming $x$ – "*the food is bad*". The green box and purple box represent an encoder and a decoder with their calculation cells (small blue boxes), respectively. $z_x$ is the latent vector of $x$ and $\oplus$ is the concatenation operator. Dashed curves represent Gumbel-Softmax, pink lines indicate transmitting relative objective losses; "*great*" with red font represents the higher weight for the target style

corresponds to $s_x^o$, with $z_x$ to form a new vector as the input of the decoder.

However, due to the scarcity of training data, the encoder sometimes cannot achieve sufficient training to make latent representations embody enough information from the original sentences. To alleviate this problem, we apply source domain $\mathcal{S}$ to append effective training data for reconstruction. However, treating all sentences of $\mathcal{S}$ as training data is impractical when there are vast sentences, whose feature distributions are different with $\mathcal{T}$, in $\mathcal{S}$ (e.g., sentiment words "*thrilling*" and "*theatrical*" are common in movie reviews but unsuited for restaurant reviews). As an alternative, we only select sentences that consist of words of $\mathcal{T}$ from $\mathcal{S}$ and the illustration is shown in Fig. 2. In this way, the increased training time can be shortened, and the useful features (determined by the same words) in $\mathcal{S}$ can be utilized effectively by the encoder. Assume $Y = \{y_i\}_{i=1}^m$ are the selected sentences from $\mathcal{S}$, since we only utilize the source domain for reconstruction training, the reconstruction loss of $y$ is given as:

$$\mathcal{L}'_{rec}(\theta_E, \theta_G, W'_s, b'_s; y) = -E_{y\sim Y}[\log p_G(y|z_y, s_y^o)] \tag{3}$$

where $s_y^o$ is the style label of $y$; $W'_s$ and $b'$ are the weight and bias matrices for building the style embeddings of $s_y^o$.

The final reconstruction loss $\mathcal{L}^*_{rec}$ with the domain adaptation is:

$$\mathcal{L}^*_{rec} = \mathcal{L}_{rec} + \lambda_{ts}\mathcal{L}'_{rec} \tag{4}$$

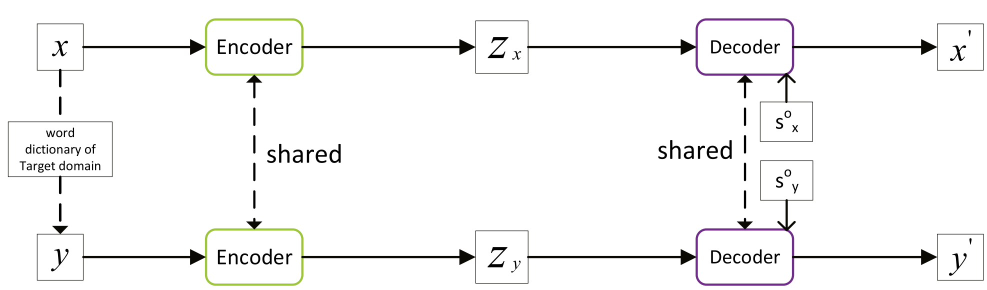
where $\lambda_{ts}$ is the balancing parameter.

### 3.2 Adversarial learning for content preservation

Previous works [20, 22, 27] attempt to obtain style-independent content representations directly from the encoder. However, in the absence of special methods and constraints, the representation will inevitably contain irrelevant style information, and the original content information will be lost in the fixed feature dimensions of $z_x$, which will lead the transferred sentence $tr(x)$ to fail in the content preservation.

To remove irrelevant style properties from $tr(x)$, we combine an adversarial network with $z_x$ to attach the point. The adversarial network adopts the idea of a generative adversarial network (GAN) [10], which contains a generator and a discriminator. In an adversarial network, the function of the discriminator is to determine whether generated samples satisfy the features of expected properties, but the generator is trained to confuse the discriminator with the generated samples. Since the discriminator and the

**Fig. 2** Illustration of reconstruction training



generator have an adversarial relationship, this can force the generator to generate high-quality data.

Concretely, we choose the encoder of the autoencoder as the generator of the adversarial network and $z_x$ as the generated target; additionally, we design a special discriminator to learn discriminating irrelevant style information in $z_x$. Since $z_x$ is a continuous variable in the feature space, we can optimize it directly to improve the quality of $tr(x)$ .

As shown in Fig. 3, the style discriminator, called $Dis_s$, provides a multilayer perceptron (MLP) to intentionally predict the style label from $z_x$. The MLP is illustrated in Fig. 4 – the content representation $z_x$ is converted to the feature value of style information by two fully connected layers, and they are calibrated to the predicted probability distribution by softmax – $y_s = MLP_{dis_s}(z_x)$. Then, $Dis_s$ is trained by minimizing the adversarial loss:

$$\mathcal{L}_{adv}(\theta_{Dis_s}) = -\sum\nolimits_{l \in labels} ds_x(l) \log y_s(l) \tag{5}$$

where $\theta_{Dis_s}$ are the parameters of $Dis_s$, $labels$ represent the two given styles $s1$ and $s2$, and $ds_x(\cdot)$ is the true style distribution (e.g., if $s_x^o$ represents $s1$, the true style distribution of $x$ is [1, 0], and [0, 1] otherwise).

Since the aim of the encoder is to generate $z_x$, which should only contain original content information, Shannon entropy is adopted as the objective loss to be transmitted to the encoder:

$$\begin{aligned}\mathcal{L}_{gen}(\theta_E) &= \mathcal{H}(y_s|z_x; \theta_{Dis_s}) \\ &= -\sum\nolimits_{l \in labels} y_s(l) \log y_s(l)\end{aligned} \tag{6}$$

where $\mathcal{H}(\cdot)$ represents Shannon entropy. The strategy is to maximize the value $\mathcal{L}_{gen}$ by optimizing $\theta_E$. As a result, the probabilities of $y_s$ all tend to $\frac{1}{2}$. In this way, $Dis_s$ cannot find a higher probability of discriminating the style label; in other words, there is no style information in $z_x$.

With the discriminator $Dis_s$ and the encoder, the overall training loss for content preservation is given as:

$$\mathcal{L}_{cp} = \mathcal{L}_{adv} - \lambda_{ag}\mathcal{L}_{gen} \tag{7}$$

where $\lambda_{ag}$ is the balancing parameter.

### 3.3 Attention model for style transfer

Including the original content information, the target style properties should be generated in transferred sentence $tr(x)$ with the generation process:

$$\begin{aligned}tr(x) = Decoder(z_x, s_x^t) &\sim p(tr(x)|z_x, s_x^t) \\ &= \prod_{i=0}^{T} p_G(tr(x)^i|tr(x)^{<i}, z_x, s_x^t)\end{aligned} \tag{8}$$

where $T$ is the max generative step of the decoder, $tr(x)^i$ is the generated word (token) at the $i^{th}$ step and $tr(x)^{<i}$ represents the generated words before the $i^{th}$ step; $s_x^t$ is represented as the style embedding for the decoder with $W_s$ and $b$ (in (2)).

Hence, we build an attention model with a bidirectional GRU (bi-GRU) [2] as the style classifier, called $Att$, to predict the style label of $tr(x)$. With training data $x$, the training process for $Att$ can be illustrated in Fig. 5 – first words $\{w_1, w_2, ..., w_L\}$ in $x$ are converted to vectors $E_x = \{e_1, e_2, ..., e_L\}$ with $e_i = W_e w_i, i \in [1, L]$, where $W_e$ is an initialized and trainable embedding matrix, then $E_x$ is fed into a bi-GRU that contains a forward GRU and a backward GRU; the forward GRU encodes $x$ from $e_1$ to $e_L$ to generate a forward hidden state of $x$ – $\overrightarrow{hs_x} = \{\overrightarrow{h_1}, \overrightarrow{h_2}, ..., \overrightarrow{h_L}\}$ and the backward GRU encodes $x$ from $e_L$ to $e_1$ to generate a backward hidden state – $\overleftarrow{hs_x} = \{\overleftarrow{h_L}, ..., \overleftarrow{h_1}\}$, where $L$ is the sum of words in $x$. Finally, the feature representation $h_i$ of word $w_i$ is acquired by concatenating $\overrightarrow{h_i}$ and $\overleftarrow{h_i}$ ($h_i = [\overrightarrow{h_i}; \overleftarrow{h_i}]$).

Since the contributions of words are not equal for the specific style properties in $x$, we apply the attention mechanism to give the importance weight for every word with the weight model. Following the attention mechanism in [33], the calculative process in the weight model is given as:

$$\begin{aligned}u_i &= tanh(W_a h_i + b_a) \\ a_i &= \frac{\exp({u_i}^T v_a)}{\sum_i \exp({u_i}^T v_a)}\end{aligned} \tag{9}$$

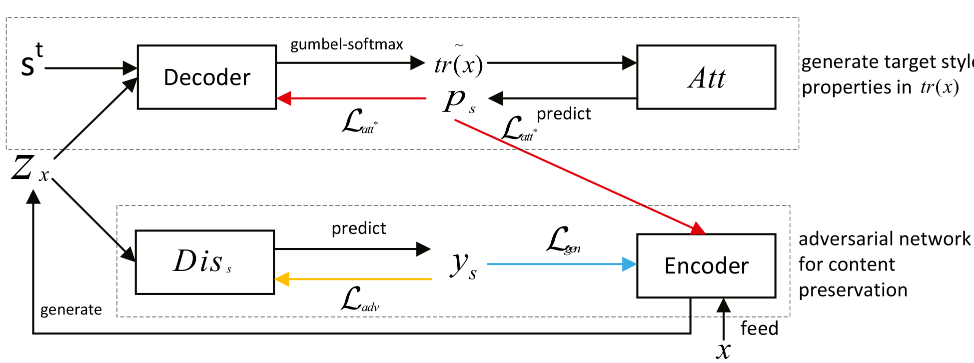



**Fig. 3** The illustration of content preservation and style transfer. $y_s$ and $p_s$ are the predicted probability distributions over labels $s1$ and $s2$ by discriminators $Dis_s$ and $Att$ respectively, and $\mathcal{L}_{att^*}$, $\mathcal{L}_{gen}$, $\mathcal{L}_{adv}$ are the training losses that are introduced in the next paragraphs

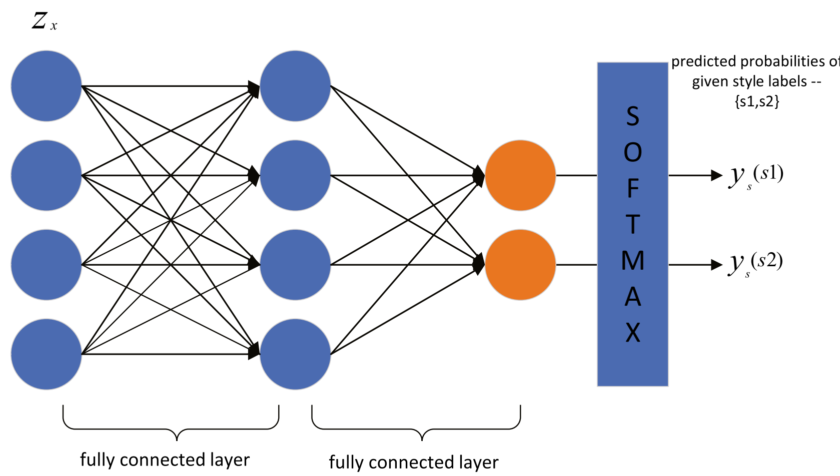



**Fig. 4** The network of the MLP for discriminating the irrelevant style information of $z_x$. The predicted probability distribution $y_s = \{y_s(s1), y_s(s2)\}$

where $W_a$ and $b_a$ are the parameters of a one-layer MLP to obtain a hidden representation $u_i$ of $h_i$; an initialized and trainable context vector $v_a$ is then applied to calculate the similarity with $u_i$, through the softmax function and the normalized output $a_i$ is regarded as the importance weight of $w_i$. Finally, the feature representation of the style properties in $x$ is given as:

$$H = \sum_{i=0}^{L} a_i h_i \tag{10}$$

The predicted probability distribution over the given style labels ($s1$ and $s2$) is $p_s = softmax(W_{att}H + b_{att})$. The cross-entropy for training the attention model is given as:

$$\mathcal{L}_{att}(\theta_{att}) = -\sum\nolimits_{l \in labels} ds_x(l) \log p_s(l) \tag{11}$$

where $\theta_{att}$ are the parameters of the attention model, $labels$ represents $\{s1, s2\}$ and $ds_x(\cdot)$ is the true style distribution of $x$.

After minimizing $\mathcal{L}_{att}$ by optimizing the relative parameters, $Att$ can be used to predict the style label of $tr(x)$. However, the discreteness of $tr(x)$ hinders the gradient propagation from $Att$ to the generator (the encoder and the decoder); as a result, $tr(x)$ cannot be optimized to append the target style properties. To find the solution method, as illustrated in Fig. 1 and Fig. 3, we use Gumbel-Softmax to obtain a continuous approximation $\widetilde{tr(x)}$ instead of $tr(x)$. The smooth process can be depicted as:

$$\widetilde{tr(x)}^i = softmax((o_i + \mathcal{N})/\tau) \tag{12}$$

where $\widetilde{tr(x)}^i$ is the distributed representation of the $i^{th}$ word in $tr(x)$, $o_i$ is the logic output of the decoder at the step of generating $\widetilde{tr(x)}^i$, $\mathcal{N}$ is the Gumbel noise, and $\tau \in (0, 1)$ is a temperature parameter to smooth the distribution. Then by the embedding matrix $W_e$, the input of the attention model is $\{W_e \widetilde{tr(x)}^i\}_{i=0}^{T}$.

To generate transferred sentences with the target style, the encoder and the decoder are adjusted by the cross-entropy:

$$\mathcal{L}_{att^*}(\theta_E, \theta_G) = -\sum\nolimits_{l \in labels} ds_{tr(x)}(l) \log p'_s(l) \tag{13}$$

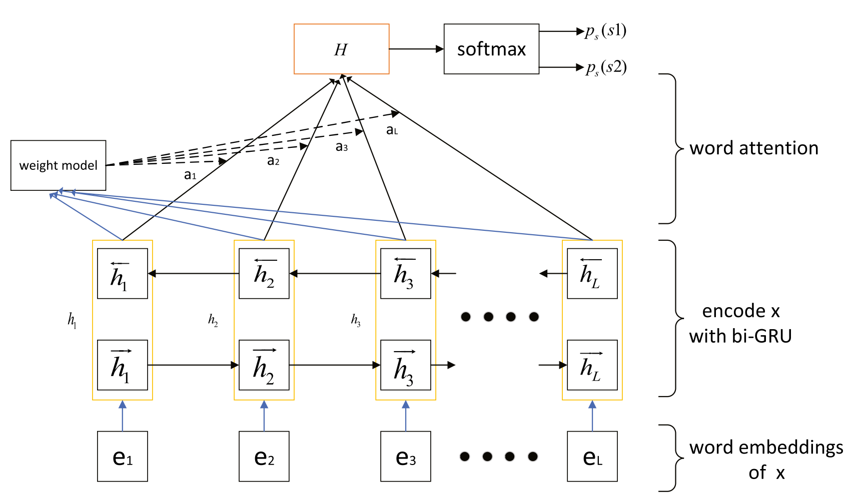



**Fig. 5** The structure of the attention model with bi-GRU for discriminating the style label $s_x^o$ from $x$. $E_x = \{e_1, e_2, ..., e_L\}$ are the word embeddings of words $\{w_1, w_2, ..., w_L\}$ in $x$. $p_s = \{p_s(s1), p_s(s2)\}$ is the predicted probability distribution over $s1$ and $s2$

where $ds_{tr(x)}(\cdot)$ is the distribution of the target style label, and $p'_s$ is the predicted probability distribution over $\{s1, s2\}$ from $\widetilde{tr}(x)$. The overall loss for style transfer with the attention model is:

$$\mathcal{L}_{st} = \mathcal{L}_{att} + \lambda_{dg}\mathcal{L}_{att^*} \tag{14}$$

where $\lambda_{dg}$ is the balancing parameter.

### 3.4 The entire training procedure

Combined with reconstruction training, adversarial learning and attention model, the training loss of the entire designed framework, which should be minimized, is given as:

$$\mathcal{L}_{wh} = \mathcal{L}^*_{rec} + \lambda_{rc}\mathcal{L}_{cp} + \lambda_{rs}\mathcal{L}_{st} \tag{15}$$

where $\lambda_{rc}$, $\lambda_{rs}$ are the balancing parameters. The training procedure for the entire framework is illustrated in Algorithm 1.

**Algorithm 1** The training procedure for the entire framework.

**Require:** The sentence sets are $X = \{x_i\}_{i=1}^n$ and $Y = \{y_i\}_{i=1}^m$. We set the balancing parameters in different training objectives and the learning rates for the optimizers of all training objectives.

1: Pretrain the autoencoder and the adversarial network for generating content representations by minimizing (4) and (5), respectively.
2: Pretrain the attention model $att$ with $X$ for discriminating style labels by minimizing (11).
3: **repeat**
4: Sample $x$ with style label $s_x^o$ from $X$, and sample $y$ with style label $s_y^o$ from $Y$.
5: Obtain the content representations $z_x$ and $z_y$ of $x$ and $y$, respectively, and proceed with reconstruction training by (4).
6: Feed $z_x$ into $Dis_s$, obtain the predicted probability distribution $y_s$ over the given style labels, train $Dis_s$ and exclude irrelevant style information from $z_x$ by minimizing (7).
7: Feed $z_x$ into the decoder with $s_x^t$, and obtain the continuous approximation $\widetilde{tr}(x)$ with (12).
8: Utilize $\widetilde{tr}(x)$ to transmit the style loss (13) to the generator, and append target style properties in $tr(x)$ by minimizing the loss.
9: **until** convergence

## 4 Experiments

To demonstrate the empirical validity and efficiency of our framework, we evaluate the quality of transferred sentences by automatic and human evaluation. We also examine the performance of our framework by comparing it to other methods in the context of limited training data and explore different cases for utilizing the source domain. Finally, we validate the contributions of different objects of the framework.

### 4.1 Datasets

The data statistics are shown in Table 1.

**Target domain** We choose three corpora – Yelp, Amazon and Captions – to evaluate our framework. The Yelp corpus (following [27]) contains the reviews of food and restaurants from the comment website Yelp, while the Amazon corpus (following [22]) covers the reviews of different commodities from Amazon. Each sentence in Yelp and Amazon is labelled with "*positive*" or "*negative*" styles (sentiments). Captions [7] are parallel sentence groups for depicting given images with "*factual*", "*romantic*" and "*humorous*" styles, and we only use sentences with "*romantic*", and "*humorous*" without utilizing the alignment information in them.

**Source domain** IMDB (following [20]) is employed as the source domain. It provides movie reviews, and each is labelled "*positive*" or "*negative*".

### 4.2 Parameter settings

The hidden dimensions of the encoder and the decoder are 500 and 700, respectively; the words that are fed to the encoder are represented by 100 dimensional vectors, which are trainable and initialized at random. The dimensions of $w$ and $b$ (symmetrically $w'$ and $b'$) in (2) are set to 200. In the attention model, we set the dimension of matrix $W_e$ for embedding words 100 dimensions (in Amazon) and 150 dimensions (in Yelp and Caption), and set the hidden dimension of the bi-GRU to 256, while $W_a$, $b_a$ and $v_a$ in

**Table 1** Statistics of source and target datasets

| Domain | Corpus | Train | Dev | Test | Words |
|---|---|---|---|---|---|
| Target | Yelp | 444101 | 63483 | 126670 | 9589 |
| | Amazon | 554997 | 2000 | 1000 | 42101 |
| | Captions | 14000 | 600 | 0 | 8916 |
| Source | Imdb | 344174 | 27529 | 27529 | 11000 |

(9) are given with 128 dimension, initialized at random. The temperature parameter $\tau$ in Gumbel-Softmax is set to 0.1.

We set the balancing parameters – $\lambda_{ag}$, $\lambda_{dg}$, $\lambda_{rc}$, $\lambda_{rs}$ and $\lambda_{ts}$ – to 1, except $\lambda_{dg} = 2$ in Amazon and $\lambda_{dg} = 1.5$ in Caption. The optimizers of all objective losses are Adam [15] with 0.0005 learning rates in Yelp and Captions, with 0.00025 learning rates in Amazon. As mentioned in Algorithm 1, the pretrain epochs are set to 5 for the Yelp corpus and 0 for the other corpora. For the Yelp corpus, the mini-batch size is 256, and the max generative step $T$ in (8) is 15; for Amazon, the mini-batch size is also 256, and the max generative step $T$ is 20; for Captions, the mini-batch size is 128, with $T = 24$. During the training, the mini-batch sizes of target domain and source domain are the same. We execute all the experiments on a Tesla v100s GPU.

### 4.3 Automatic evaluation metrics

We evaluate transferred sentences with automatic evaluation and human evaluation. In the automatic evaluation, we analyze three aspects: style transfer accuracy, content preservation, and domain accuracy (The automatic evaluation results in all experiments are generally the best results considering all the indexes of automatic evaluation when the training losses in given frameworks basically reach convergence.):

**Style transfer accuracy (Sac)** We combine TextCNN [14] with word embedding from pretrained 300D GloVe vectors [26] to design style classifiers for the Yelp corpus (up to 0.97 accuracy), Amazon corpus (up to 0.82 accuracy) and Captions corpus (up to 0.98 accuracy). Style transfer accuracy is given as:

$$Sac = \frac{TS^{tr}}{TS} \tag{16}$$

where $TS^{tr}$ represents the sum of transferred sentences with the target style label and $TS$ represents the sum of transferred sentences.

**Content preservation** Since the same content can be described in different ways under different styles, the performance for content preservation is difficult to evaluate. Hence, we adopt three indexes to measure content similarity between $tr(x)$ (transferred sentence) and $x$ (original sentence):

- *Word Overlap* (**WO**): Following [13], we adopt Word Overlap as a simple yet effective metric for content preservation. It can be computed as:

$$\mathrm{WO} = \frac{count(ws_o \cap ws_t)}{count(ws_o \cup ws_t)} \tag{17}$$

where $ws_o$ and $ws_t$ are the word sets (uni-grams) of $x$ and $tr(x)$, respectively, and $count(\cdot)$ indicates the sum of the words. The stopwords and the given style words (often occur in sentences with one style label and are rare in another style label) are excluded from the word sets since we only consider content information.
- *Bilingual Evaluation Understudy* (**BLEU**) [24]: In contrast to Word Overlap, Bleu builds an n-gram model between $tr(x)$ and $x$, then utilizes the same n-grams to evaluate their similarity. The calculation is given roughly by 5

$$\mathrm{BLEU} = \mathrm{BP} \times \exp\left(\sum_{i=1}^{N} \frac{1}{N} \log p_n\right) \tag{18}$$

where BP is the brevity penalty for the word sums between $x$ and $tr(x)$, and $p_n$ is the ratio between the same n-gram counts and the n-gram counts in $tr(x)$ ($N$ is generally set to 3; except in Fig. 6, BLEU scores are represented as percentages).
- Cosine Similarity [6] (**CS**): This is equivalent to the cosine distance between the sentence embeddings of $x$ and $tr(x)$. The sentence embedding is given by concatenating the min, mean and max word embeddings (excluding the embedding of specific style words like in Word Overlap, and the embeddings are obtained from pretrained 100D GloVe vectors).

**Domain accuracy (Dac)** The design of the domain classifier is the same as that of the style classifier, and the accuracies for Yelp and Captions all reach 0.99. It is given by:

$$Dac = \frac{TD^{tr}}{TD} \tag{19}$$

where $TD^{tr}$ represents the sum of transferred sentences with the target domain label and $TD$ represents the sum of transferred sentences.

### 4.4 Experimental results and analyses

#### 4.4.1 Experiment I: Automatic evaluation

We compare our framework with other frameworks on automatic evaluation as seen in Table 2, and some sampled transferred sentences are given in Table 3. We obtain the following conclusions from these results:

- Observing that CrossAlign obtains the worst Sac scores of 0.792 and 0.822 on Yelp and Captions respectively, we conclude that it is the vanishing gradient problem of a conventional adversarial network that causes the failure of generating target style properties; furthermore, the lack of explicit constraints for content preservation leads to the loss of original content information, as demonstrated by the lowest scores of content preservation on Yelp and Amazon.

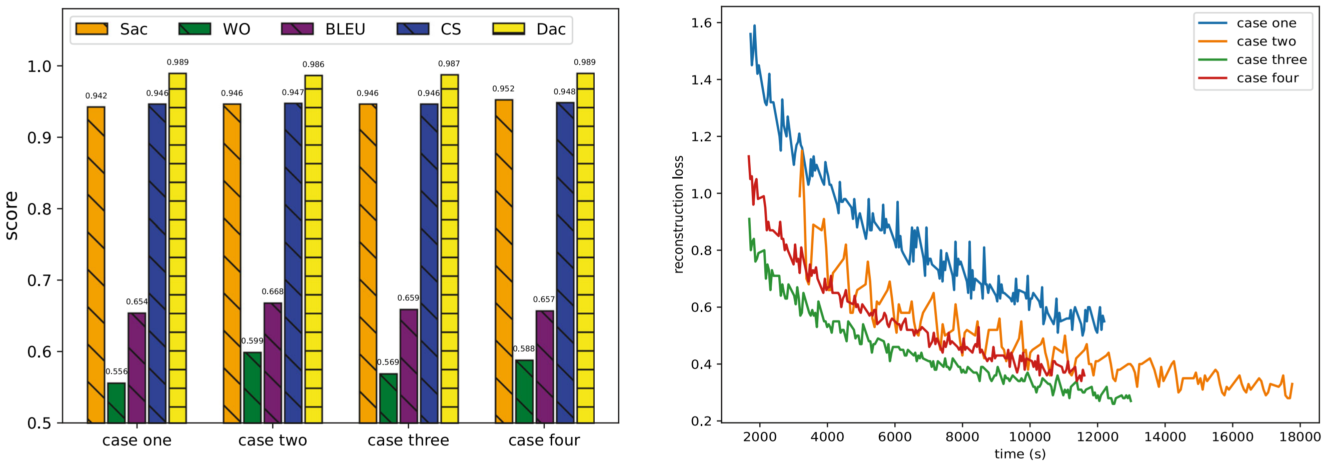



**Fig. 6** The automatic evaluation (left) and the reconstruction loss of mini-batches over time (right) for different cases on the Yelp corpus (due to some factors including the random orders of mini-batches and the random seeds in the parameters, the reconstruction losses have an error margin of 0.08 from observation; however, the float does not cause any influence for discriminating the changing tendency)

- For Reinforce, the Sac score on Amazon is much lower than on Yelp and Captions, and meanwhile we find that the attention-based neutralization model fails to discriminate specific style words which affects the direction of rewards of reinforcement learning, hindering the decoder in generating target style properties; contrarily, the decoder only copies the original sentences with high scores of content preservation and the unsuccessfully transferred samples on Amazon. It tends to completely lose the content information of Captions with only a 0.049 WO score and 8 BLEU score, and the main reason can be attributed to the fact that reinforcement learning cannot be conducted effectively from the limited training data of Captions without enough prior knowledge.
- In light of the relatively high Sac scores, DiseRe seems to capture the representation vectors of different style labels well through disentanglement representation learning. However, the quite low BLEU scores on all corpora indicate that it ignores the content preservation for sentences. To this point, we determine that VAE in it forces the latent representations of sentences following a prior distribution (normal distribution), which allows the decoder to generate multiple samples based on the distribution; however this manner allows the latent representations not to focus on carrying sufficient information of the origianl sentences; meanwhile, it utilizes fully connected layers to align the content representation vectors with the BOW features of sentences, which is equivalent to original content information, leading to the loss of the position information of words since the BOW features are unordered; moreover, the determinations of the BOW features, which depend on excluding high frequency words about style labels, are always inaccurate.
- Compared with [13, 27, 32], Controlgen achieves a better performance as shown by the comprehensive scores on all corpora, since the self framework does not contain evident shortcomings. Observing the indexes of content preservation and the saved content words of samples on Amazon, DAST exhibits better performance on content preservation, which can be attributed to domain adaptation learning. In obtaining the most superior scores on Yelp, our framework exceeds other frameworks in style transfer and content preservation. It is not difficult to determine that beside domain adaptation learning, the attention model assigns the importance weights of words for the target style to drive the decoder to better generate target style properties better; the adversarial network, which filters the irrelevant style information from content vectors, further guarantees the quality of transferred sentences.
- Unusually, all of the frameworks fail to transfer the sentences in the Amazon corpus: we analyse the classifier with 0.82 accuracy causes inaccurate results of Sac, and similarly, it is also difficult for the style discriminators in all frameworks to discriminate the style information of sentences, and the decoder cannot be optimized to generate target style properties in sentences. Another inference is that the vast amount of words in Amazon (as seen in Table 2) causes the difficulty of learning sentence features, leading to the loss of the content information of sentences.

#### 4.4.2 Experiment II: Human evaluation

Since automatic evaluation can only obtain scores from fixed calculations, we implement human evaluation to evaluate the quality of transferred sentences. We choose 100

**Table 2** Automatic evaluation results of different methods on the test data of all corpora (the bold fonts represent the highest scores of the corresponding indexes)

| Framework (Method) | Yelp | | | | | Amazon | | | | | Captions | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Sac | WO | BLEU | CS | Dac | Sac | WO | BLEU | CS | Dac | Sac | WO | BLEU | CS | Dac |
| CrossAlign [27] | 0.792 | 0.156 | 15.4 | 0.855 | - | 0.709 | 0.10 | 8.75 | 0.899 | - | 0.822 | 0.163 | 23.8 | 0.916 | - |
| Reinforce [32] | 0.833 | 0.312 | 39.5 | 0.904 | - | 0.473 | **0.342** | 53.6 | 0.934 | - | **0.979** | 0.049 | 8 | 0.898 | - |
| Controlgen [11] | 0.945 | 0.489 | 58.5 | 0.934 | - | **0.816** | 0.144 | 27.4 | 0.912 | - | 0.942 | 0.141 | 18.4 | 0.926 | - |
| DiseRe(VAE) [13] | 0.915 | 0.333 | 13.3 | 0.893 | - | 0.804 | 0.293 | 9.1 | 0.898 | - | 0.877 | 0.154 | 12.4 | 0.924 | - |
| DAST[20] | 0.934 | 0.522 | 63.3 | 0.938 | 0.987 | 0.752 | 0.218 | 47.6 | 0.924 | 0.981 | 0.892 | 0.241 | 25.7 | 0.934 | 0.992 |
| Ours | **0.952** | **0.588** | **65.7** | **0.948** | **0.989** | 0.769 | 0.286 | **56.7** | **0.931** | **0.983** | 0.900 | **0.248** | **32.1** | **0.940** | **0.994** |

sentences from Yelp, and referring to [13], the annotators are asked to evaluate the corresponding transferred sentences with $1-5$ scores with respect on the aspects of transfer strength, content similarity, and language quality which pertains to language fluency and grammar. The results are presented in Table 4.

Observing the highest scores of transfer strength and content similarity for our framework, we analyze their relevance with the scores of style transfer accuracy (Sac) and content preservation in Table 2. And the score order in content similarity is consistent with those in BLEU and CS from lowest to highest, which also proves the relevance between content similarity and the indexes of content preservation. However, the level relationship between DiseRe and DAST in transfer strength is different from that in Sac, and we infer that the reasons for this are as follows: 1. the sampled sentences are random – they may contain more failure examples for DAST; 2. the index Sac is inaccurate for some transferred sentences – note that on Yelp, the sentence "*the food is excellent, and the service is exceptional!*" carries "*positive*" style (sentiment), while the transferred sentence "*the food is bad, and I like the service!*" surely receives a much lower transfer strength score, although it would be classified as having "*negative*" style since the style classifier focuses on "*the food is bad*" and ignores the second half "*I like the service*". We find that the automatic evaluation indexes do not have evident relevance with language quality; however, the better performance for automatic evaluation results tends to produce a higher score for language quality, as demonstrated by the 3.90 score of DAST and 3.85 score of our framework. On the other hand, we should improve the automatic evalution indexes to evaluate sentences roundly in future work.

In general, compared with other frameworks, our framework achieve the best performance of automatic evaluation and human evaluation; however, the scores are not high. In the description of the annotators, our framework fails to transfer long and complex sentences without loss the content information. We analyze the constituent structures of long sentences which cause the difficulty of feature learning and the error of the extrated content information; moreover, the attention model are incompetent to generate target properties with the inaccurate content information. We think we should utilize part-of-speech and the position information of words to explore more sentence features and add relative mechanisms to capture the content information accurately in further work.

#### 4.4.3 Experiment III: Limit the training data of the target domain

In this experiment, we execute the automatic evaluation of different methods under 10% training data from the Yelp

**Table 3** Transferred samples on Yelp, Amazon and Captions (the transferred parts are colored)

| | Yelp (positive-to-negative) | Yelp (negtive-to-positive) |
|---|---|---|
| Original | the food here is amazing and tastes fresh in every bite | you will leave frustrated and unhappy with the food and the service |
| CrossAlign | the food here is and num stars are the same time | we will recommend their service and the service and the service |
| Reinforce | the food here is wrong and tastes fresh in every reply | you will leave welcome and pleased with the quick and excellent itself |
| Controlgen | the food here is nasty and tastes mediocre in every bite | you will leave was and pleasant with the food and the service |
| DiseRe | the food is not fresh and tastes like it was a little salty | i am so happy with the service and food |
| DAST | the food here is dreadful and tastes fresh in every bite | you will leave exceptional and consistent with the food and the service |
| Ours | the food here is lousy and tastes lukewarm in every bite | you will leave impressed and happy with the food and the service |
| | Amazon (positive-to-negative) | Amazon (negative-to-positive) |
| Original | durable and seems similar quality to apple products | in this case , it is a big disappointment for num-extend |
| CrossAlign | great product to be very good and made for me | for this case , it is a great for my kitchen |
| Reinforce | and creates to be edible n in products | in this case , it is a big disappointment for me |
| Controlgen | graphics are useless to poor quality players | in this case , it is a better quality for a minute |
| DiseRe | but i am not sure what i was looking for | i have a big num extend and it is a big deal , but it is a big deal |
| **DAST** | durable , seems similar quality to customer service products | in this case , it is a big appliance for num-extend |
| Ours | useless and seems worthless design to nutritional products | in this case , it is a big charge for num-extend |
| | Captions (romantic-to-humorous) | Captions (humorous-to-romantic) |
| Original | these racing dogs going for gold | a dog is galloping on the beach , in order to snatch a frisbee |
| CrossAlign | because he specifically asked to be the ball | a dog is running on the beach with his friends |
| Reinforce | a dog is running to a water | a man is running in a red dog is enjoying the beach of his lover |
| Controlgen | three dogs run together like gladiators | a dog is running on the beach in the field of a loving owner |
| DiseRe | two dogs are running through the snow to find bones | a dog runs through the snow , hoping to win the race |
| DAST | many dogs go for supremacy | a dog is walking on the grass , in order to catch a frisbee |
| Ours | several other dogs run for supremacy | a dog is running on the beach , to catch a joy in victory |

corpus, and the results are shown in Table 5 (the identified words also should be adjusted for 10% training data).

We observe that the performances of all methods exhibit serious reductions on content perservation with limited training data, especially Reinforce [32], which only achieve a 0.1 WO score since reinforcement learning is unable to utilize limited training data to obtain a correct policy for high rewards.

DAST [20] and our method still achieve absolute advantages with respect to the scores of WO, BLEU and CS; meanwhile, the 0.983 and 0.985 Dac scores indicate that domain adaptation does not introduce unique features of the source domain. These results indicate that domain adaptation learning for reconstruction can promote the capacity of the framework to capture and preserve content in formation.

#### 4.4.4 Experiment IV: Different cases for utilizing source domain

To prove that it is effective to select the sentences, consisting of the words of the target domain, in the source domain for reconstruction training, we explore the automatic evaluation performances for different cases on the Yelp corpus, and record the reconstruction losses of mini-batches of Yelp over time; to describe briefly, we intercept the loss results from 7th epoch to the epoch which corresponds to the automatic evaluation result (case one – 31st epoch, case two – 29th epoch, case three – 32nd epoch, case four – 29th epoch):

**case one** do not employ source domain for reconstruction training.

**case two** discriminate all words of the source domain to add all sentences for reconstruction training.

**Table 4** Human evaluation on the transferred sentences of the Yelp test data (the bold fonts represent the highest scores of the corresponding indexes)

| Framework | Transfer Strength | Content Similarity | Language Quality |
|---|---|---|---|
| CrossAlign [27] | 2.75 | 2.65 | 3.60 |
| Control [27] | 3.90 | 3.55 | 3.75 |
| Reinforce [32] | 3.70 | 3.45 | 3.65 |
| DiseRe(VAE) [13] | 4.25 | 3.05 | 3.75 |
| DAST [20] | 4.10 | 3.70 | **3.90** |
| Ours | **4.35** | **3.85** | 3.85 |

**Table 5** Automatic evaluation on the Yelp test data, trained under 10% training data of Yelp corpus (the bold fonts represent the highest scores of the corresponding indexes)

| Framework(method) | Sac | WO | BLEU | CS | Dac |
|---|---|---|---|---|---|
| CrossAlign | 0.848 | 0.110 | 11.2 | 0.863 | – |
| Reinforce | 0.927 | 0.100 | 11.6 | 0.870 | – |
| Controlgen | 0.948 | 0.308 | 36.3 | 0.911 | – |
| DiseRe(VAE) | 0.881 | 0.245 | 12.9 | 0.881 | – |
| DAST | **0.956** | 0.427 | 53.3 | 0.920 | 0.983 |
| Ours | 0.940 | **0.458** | **54.9** | **0.931** | **0.985** |

**case three** only discriminate the words of the target domain for the sentences in the source domain, and treat other words as "*UNK*".

**case four** select the sentences, consisting of the words of the target domain, in the source domain.

With the visualized comparson in the histogram and the curve graph, we find that domain adaptation learning can effectively help the reconstruction training with the lower reconstruction losses at the same time, which indicates that the training data from the source domain allow the encoder and the decoder to learn more useful features for reconstructing data; the performances for content preservation are also promoted since the content representations of sentences can be obtained accurately by reconstruction. Specifically, the Dac score of case one is not 1, and we determine that this is attributed to the deviation of the classifier and some failure in transferring sentences. Case two produces the highest WO and BLEU scores – 0.599 and 0.668, which can be attributed to the use of the most training data for reconstruction training. However, this means that the framework should discriminate significantly more different words, and then the encoder and the decoder learn and optimize the embeddings of the added words in the reconstruction, which wastes considerable time as observed in Table 6. Case three treats the extra words from the source domain as "*UNK*", and in this way the training time is shortened while maintaining the lowest reconstruction loss; however, the word "*UNK*" appears 18 times in the transferred sentences. Therefore, we choose case four for our framework since the performance for content preservation is close to those of case two and case three, and the training time is the shortest, especially when the source domain contains a large number of different words with the target domain.

#### 4.4.5 Experiment V: Ablation test of our framework

In this experiment, we utilize the autoencoder as the baseline to research the contributions of domain adaptation, adversarial learning and attention model on Yelp corpus. To highlight their advantages, we implement the ablation test for our framework with only 10% training data of the target domain.

As shown in Table 6, our approach obtains the best comprehensive performance with respect to all of the objectives; on the contrary, only adopting the autoencoder model just learns copying original sentences with 0.225 transfer accuracy. In addition, the attention model improves the transfer accuracy from 0.225 to 0.964, which proves its effectiveness for style transfer. With appended adversarial learning, the performance for content preservation achieves a promotion versus the variant – "*autoencoder + attention model*". Relatively, the domain adaptation learning makes a greater contribution to content preservation, especially improving the Bleu from 33.5 to 48.2; moreover, the slight variation in Dac scores of all variants further proves the effectiveness of domain adaptation.

**Table 6** Ablation tests evaluated on the Yelp test data, trained under 10% training data of the Yelp corpus (the bold fonts represent the highest scores of the corresponding indexes)

| Variants | Sac | WOx | BLEU | CS | Dac |
|---|---|---|---|---|---|
| autoencoder model | 0.225 | **0.672** | **67.3** | **0.963** | **0.989** |
| +attention model | **0.964** | 0.299 | 33.5 | 0.908 | 0.986 |
| +attention model+ domain adaptation | 0.956 | 0.386 | 48.2 | 0.914 | **0.989** |
| +attention model+ adversarial learning | 0.956 | 0.330 | 37.2 | 0.913 | 0.983 |
| +attention model+ domain adaptation + adversarial learning | 0.940 | 0.458 | 54.9 | 0.931 | 0.985 |

The same parameters of the all variants are set to be unanimous

## 5 Conclusion

In this paper, we adopt domain adaptation to help an autoencoder develop the capacity for reconstructing data by only selecting sentences from the source domain that are composed by the words of the target domain. Moreover, we adopt adversarial learning to allow latent representations of sentences carrying more style-independent content information. Then, the attention-based style classifier utilizes attention mechanism to allow the generator to concentrate on generating words that are given higher importantce weights for target style. Through relevant experiments, our framework demonstrates obvious advantages over previous advanced methods in the context of non-parallel and limited training data.

**Acknowledgements** The authors would like to appreciate the editors and the reviewers for their constructive suggestions. And the creation of the paper cannot leave with the excellent papers on deep learning and their codes.

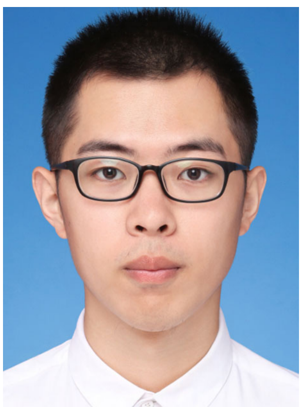
This work is supported by the Science and Technology Plan of Yunnan Province of China under grants 2014AB016.

## References

1. Arjovsky M, Bottou L (2017) Towards principled methods for training generative adversarial networks. In: International Conference on Learning Representations
2. Bahdanau, D, Cho, K, Bengio, Y (2014) Neural machine translation by jointly learning to align and translate. In: International Conference on Learning Representations
3. Chen, X, Duan, Y, Houthooft, R, Schulman, J, Sutskever, I, Abbeel, P (2016) Infogan: interpretable representation learning by information maximizing generative adversarial nets. In: Advances in neural information processing systems, pp 2180–2188
4. Cho, K, Van Merrienboer, B, Gulcehre, C, Bahdanau, D, Bougares, F, Schwenk, H, Bengio, Y (2014) Learning phrase representations using rnn encoder-decoder for statistical machine translation. In: Empirical methods in natural language processing, pp 1724–1734
5. Dong, H, Neekhara, P, Wu, C, Guo, Y (2017) Unsupervised image-to-image translation with generative adversarial networks. In: IEEE Conference on computer vision and pattern recognition
6. Fu, Z, Tan, X, Peng, N, Zhao, D, Yan, R (2018) Style transfer in text: Exploration and evaluation. In: AAAI Conference on Artificial Intelligence
7. Gan, C, Gan, Z, He, X, Gao, J, Deng, L (2017) Stylenet: Generating attractive visual captions with styles. In: IEEE Conference on computer vision and pattern recognition, pp 955–964
8. Gatys, LA, Ecker, AS, Bethge, M (2016) Image style transfer using convolutional neural networks. In: IEEE Conference on computer vision and pattern recognition, pp 2414–2423
9. Gezici, G, Yanikoglu, B, Tapucu, D, Saygın, Y (2015) Sentiment analysis using domain-adaptation and sentence-based analysis. In: Advances in Social Media Analysis, Springer, pp 45–64
10. Goodfellow, I, Pouget-Abadie, J, Mirza, M, Xu, B, Warde-Farley, D, Ozair, S, Courville, A, Bengio, Y (2014) Generative adversarial nets. In: Advances in neural information processing systems, pp 2672–2680
11. Hu, Z, Yang, Z, Liang, X, Salakhutdinov, R, XingvEP (2017) Toward controlled generation of text. In: International Conference on Machine Learning, pp 1587–1596
12. Jang, E, Gu, S, Poole, B (2016) Categorical reparameterization with gumbel-softmax. arXiv:16110114
13. John, V, Mou, L, Bahuleyan, H, Vechtomova, O (2019) Disentangled representation learning for non-parallel text style transfer. In: Association for Computational Linguistics, pp 424–434
14. Kim, Y (2014) Convolutional neural networks for sentence classification. In: Empirical methods in natural language processing, pp 1746–1751
15. Kingma, D, Ba, J (2014) Adam: A method for stochastic optimization. In: International Conference on Learning Representations
16. Kingma, DP, Welling, M (2014) Auto-encoding variational bayes. In: International Conference on Learning Representations
17. Kullback, S, Leibler, RA (1951) On information and sufficiency. The annals of mathematical statistics 22(1):79–86
18. Kusner, MJ, Hernandezlobato, JM (2016) Gans for sequences of discrete elements with the gumbel-softmax distribution. arXiv preprint arXiv:161104051
19. Lamb, AM, Goyal, AGAP, Zhang, Y, Zhang, S, Courville, AC, Bengio, Y (2016) Professor forcing: A new algorithm for training recurrent networks. In: Advances in neural Information Processing Systems, pp 4601–4609
20. Li, D, Zhang, Y, Gan, Z, Cheng, Y, Brockett, C, Dolan, B, Sun, M (2019) Domain adaptive text style transfer. In: International joint conference on natural language processing, pp 3302–3311
21. Li, J, Monroe, WS, Shi, T, Jean, S, Ritter, A, Jurafsky, D (2017) Adversarial learning for neural dialogue generation. arXiv preprint arXiv:170106547
22. Li, J, Jia, R, He, H, Liang, P (2018) Delete, retrieve, generate: A simple approach to sentiment and style transfer. In: North american chapter of the association for computational linguistics, vol 1, pp 1865–1874
23. Liu, M, Breuel, TM, Kautz, J (2017) Unsupervised image-to-image translation networks. In: IEEE Conference on computer vision and pattern recognition
24. Papineni, K, Roukos, S, Ward, T, Zhu, WJ (2002) Bleu: a method for automatic evaluation of machine translation. In: Association for Computational Linguistics, Association for Computational Linguistics, pp 311–318
25. Peng, N, Dredze, M (2016) Multi-task domain adaptation for sequence tagging. arXiv:160802689
26. Pennington, J, Socher, R, Manning, CD (2014) Glove: Global vectors for word representation. In: Empirical methods in natural language processing, pp 1532–1543
27. Shen, T, Lei, T, Barzilay, R, Jaakkola, T (2017) Style transfer from non-parallel text by cross-alignment. In: Advances in neural information processing systems, pp 6830–6841
28. Sutton, RS, McAllester, DA, Singh, SP, Mansour, Y (2000) Policy gradient methods for reinforcement learning with function approximation. In: Advances in neural information processing systems, pp 1057–1063
29. Wen, T, Gasic, M, Mrksic, N, Su, P, Vandyke, D, Young, S (2015) Semantically conditioned lstm-based natural language generation for spoken dialogue systems. In: Empirical methods in natural language processing, pp 1711–1721
30. Wen, T, Gasic, M, Mrksic, N, Rojasbarahona, LM, Su, P, Vandyke, D, Young, S (2016) Multi-domain neural network language generation for spoken dialogue systems. arXiv:160301232
31. Wu, Y, Schuster, M, Chen, Z, Le, QV, Norouzi, M, Macherey, W, Krikun, M, Cao, Y, Gao, Q, Macherey, K, et al (2016) Google's neural machine translation system: Bridging the gap between human and machine translation. arXiv:160908144

32. Xu, J, Sun, X, Zeng, Q, Zhang, X, Ren, X, Wang, H, Li, W (2018) Unpaired sentiment-to-sentiment translation: A cycled reinforcement learning approach. In: Association for Computational Linguistics, vol 1, pp 979–988
33. Yang, Z, Yang, D, Dyer, C, He, X, Smola, AJ, Hovy, E (2016) Hierarchical attention networks for document classification. In: North american chapter of the association for computational linguistics, pp 1480–1489
34. Yi, Z, Zhang, H, Tan, P, Gong, M (2017) Dualgan: Unsupervised dual learning for image-to-image translation. In: IEEE Conference on computer vision and pattern recognition
35. Yu, L, Zhang, W, Wang, J, Yu, Y (2017) Seqgan: Sequence generative adversarial nets with policy gradient. In: AAAI Conference on Artificial Intelligence, pp 2852–2858
36. Zhu, JY, Park, T, Isola, P, Efros, AA (2017) Unpaired image-to-image translation using cycle-consistent adversarial networks. In: International Conference on Computer Vision, pp 2242–2251





**Mingxuan Hu** received his Bachelor degree from the School of Information Engineering, Jiangxi University of Technology in 2018.

Now, he is pursuing his master degree in the School of Information, Yunnan University. His research interests include natural language processing, complex network analysis and knowledge graph.

**Min He** received her B.Sc. and M.Sc. degrees in computer application technology both from Liaoning Technical university, in 1998 and 2001, respectively, obtained Ph. D. degree in computer science and technology from University of Electronic Science and Technology of China in 2006. Now, she is an associate professor and master student supervisor in Yunnan University. Her main research interests include intelligent information process, social network analysis and embedded system application.