Text Style Transferring via Adversarial Masking and Styled Filling

Jiarui Wang[1], Richong Zhang[1, 2 *], Junfan Chen[1], Jaein Kim[1], Yongyi Mao[3]
[1]SKLSDE, Beihang University, Beijing, China
[2]Zhongguancun Laboratory, Beijing, China
[3] School of Electrical Engineering and Computer Science, University of Ottawa, Canada
{wangjr,zhangrc,chenjf}@act.buaa.edu.cn
jaein@buaa.edu.cn,ymao@uottawa.ca

Abstract

Text style transfer is an important task in natural language processing with broad applications. Existing models following the masking and filling scheme suffer two challenges: the word masking procedure may mistakenly remove unexpected words and the selected words in the word filling procedure may lack diversity and semantic consistency. To tackle both challenges, in this study, we propose a style transfer model, with an adversarial masking approach and a styled filling technique (AMSF). Specifically, AMSF first trains a mask predictor by adversarial training without manual configuration. Then two additional losses, i.e. an entropy maximization loss and a consistency regularization loss, are introduced in training the word filling module to guarantee the diversity and semantic consistency of the transferred texts. Experimental results and analysis on two benchmark text style transfer data sets demonstrate the effectiveness of the proposed approaches.

1 Introduction

Stylistic attributes of natural language text have intrigued researchers in natural language processing (NLP) for a long time. Text style transfer aims to convert the text into a target style while preserving the content of the source text, which has broad industrial applications such as computer-aided writing (Klahold and Fathi, 2020) and advertising systems (Jin et al., 2020). Due to the lack of parallel data, text style transfer is usually treated as an unsupervised learning task.

One of the mainstream approaches is to exploit the Autoencoder to disentangle the style and content representation of the source text, then use the decoder to generate a transferred text with the disentangled content representation and the target style (Fu et al., 2018; John et al., 2018; Shen et al., 2017; Nangi et al., 2021). The weakness of such a sequence-to-sequence manner is that the model usually performs poorly on content preservation because the intrinsic of generative models bring a significant difference between source and target text. Another branch of models follows the masking and filling scheme (Devlin et al., 2018; Xu et al., 2018; Sudhakar et al., 2019; Wu et al., 2019), which first explicitly masks out the stylistic words of the source style in the text and then replaces them with words from the target style. These models usually use either average attention scores (Bahdanau et al., 2014) or the higher frequency ratio as criteria to select the stylistic words in the word masking procedure. However, these masking approaches may mistakenly remove the non-stylistic words when the attention distribution is insignificant (Lee et al., 2021).

When filling the masked words, previous works utilize BERT (Devlin et al., 2018) to fill the masked positions with the supervision from the style classifier (Wu et al., 2019) or retrieve words with similar attributes (Li et al., 2018; Sudhakar et al., 2019) from the target domain. The limitation of these filling approaches is that the diversity and semantic consistency of filled words are not guaranteed. We argue that diversity and semantic consistency are significant in text style transfer. An ideal style transfer model should replace source stylistic words with their semantically-consistent counterpart in the target domain. For example, in sentiment transfer task, it is more reasonable to transfer "fast" to "slow" rather than "poor", and "worst" is preferred to be transferred from "best" instead of "good".

To overcome the imprecise word masking and the limitation of lacking diversity and consistency in word filling, in this study, we propose an adversarial masking approach and a styled filling technique for the text style transfer task. Specifically, to improve the word masking quality, we introduce an adversarial gating strategy in training the mask predictor. The mask predictor is trained as a gen-

[*] Corresponding author

Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, pages 7654–7663
December 7-11, 2022 ©2022 Association for Computational Linguistics

erator in an adversarial way with a discriminator identifying the source style from the masked sequence. In addition, the number of masked words is also restricted to prevent the mask predictor from sacrificing non-stylistic words. The mask predictor trained with an adversarial gating strategy is experimentally shown to be effective in selecting stylistic words and flexible in controlling of masked ratio without additional statistical or manually labelled information.

To improve the diversity and consistency of the filled words, we develop a styled filling approach based on BERT Masked Language Model to predict the substituting vocabulary distribution. This approach introduces an entropy-based diversity loss and a semantic-based consistency loss to encourage the word-level language diversity and semantic consistency. Language diversity of styled filling is realized by punishing the low diversity of generated target stylistic words. The motivation to incorporate consistency loss is that source stylistic words usually share similar context with their antonyms and have closer word embedding vectors compared to other words in the target domain. The designed consistency loss can shorten the distance between transferred word embedding and source stylistic word embedding as the guidance for our model to predict the antonyms.

Empirical studies show that our model AMSF outperforms existing approaches in terms of the overall score of content preservation and transfer accuracy. In addition, further comprehensive analysis of the diversity and the semantic consistency of generated text confirms the effectiveness of the proposed approach.

In summary, the contributions of this study are as follows:

- We introduce an adversarial masking approach that can more accurately mask out stylistic words without additional statistical information and automatically generate a gate sequence without introducing extra manually set rules.

- We present a word filling method that leverages the semantic distance and utilizes the vocabulary entropy to improve the semantic consistency and language diversity of generated text.

- The empirical studies on two benchmark text style transfer data sets demonstrate the superiority of our proposed approach AMSF.

2 Related Work

Due to the scarcity of parallel data in text style transfer, the mainstream recent research has regarded the task of text style transfer as an unsupervised learning task.

In recent works, one category of the methods aims to first implicitly filter out the style-related words in the text by learning a latent representation of content and style and then generating new text of the target style. Hu et al. (2017) learns the latent representation of text utilizing variational autoencoder (VAE) (Kingma and Welling, 2013) framework and utilizes a style classifier to learn a style attribute vector. Fu et al. (2018) leverages an adversarial network to train a content encoder, the encoded content vector is then transferred by the style-specific decoder. John et al. (2018) adopts VAE to separate the style representation and content representation, with the decoder directly taking the concatenation of the encoded content representation and target style representation as input.

There is a branch of study that performs text style transfer task without disentangling the source text into the style and content. For instance, the dual structure, which learns the source-to-target and target-to-source mappings, is applied to achieve the goal. One of the works(Li et al., 2020) trains the transfer models in the way of training de-noising autoencoders (DAE) (Vincent et al., 2008) with noisy text using neighbourhood sampling approach and the models are trained cyclically to reconstruct origin text from noisy input.

Another mainstream of the works attempts to explicitly replace the keywords and generate a text of the target style. Identifying style-related words in the given text is of great importance in this method. While Li et al. (2018) selects the stylistic words according to their frequency in text from a specific style, most early works utilize the attention mechanism as an indicator of stylistic words. Xu et al. (2018) and Wu et al. (2019) take the average attention score as the threshold for filtering out the style-related words. Sudhakar et al. (2019) calculates attention on each (layer, head) pair of Transformer (Vaswani et al., 2017) and selects one pair with the highest score. Lee et al. (2021) also leverages attention score as style information, but the reversed attention is taken as weights on content

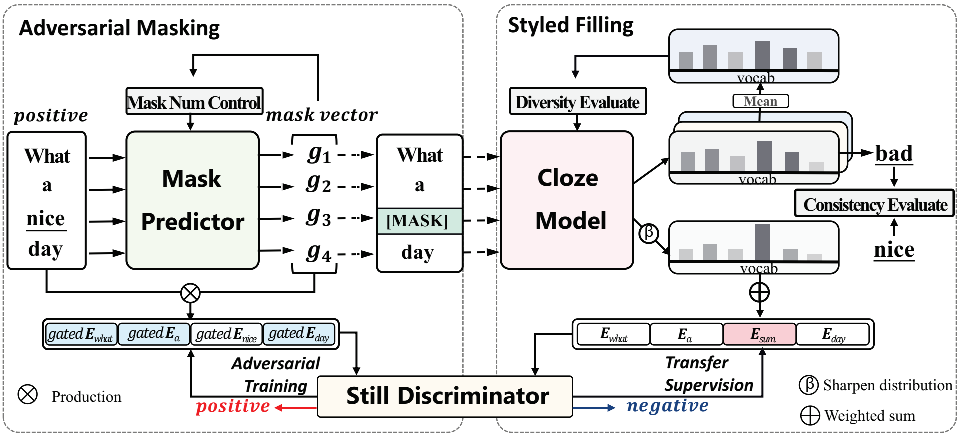


Figure 1: The overall AMSF architecture. Mask Predictor G is trained in an adversarial way and generates a mask vector g. Bert-Based Cloze Model C takes masked sequence $\tilde{X}$ as input, and fills in words in target style.

representation instead of masking words directly in this work.

3 Approach

3.1 Task Definition

Text style transfer aims to convert the style of the given text, e.g., from the positive sentiment to the negative sentiment. Formally, we are given two non-parallel corpus $\mathcal{D}_x = \{X_i\}$ and $\mathcal{D}_y = \{Y_j\}$, with corresponding styles s_x and s_y respectively. Each X_i or Y_j represents a text in the corpus. The goal of text style transfer is to train a style transfer model that enables to transfer a text from *source style* to *target style* while preserving its original content. Namely, it should convert a text X_i with style s_x to style s_y or conversely transfer a text Y_j with style s_y to style s_x.

3.2 Model Overview

We propose a model that converts the style of a text by identifying the stylistic words and explicitly replacing them. To that end, we build a three-phase training procedure. The first phase is to train a mask predictor G in both corpora that produce a mask vector indicating the positions of stylistic words in a text which should be replaced by the opposite-style words. The second phase is to pre-train a cloze model C using a source-style corpus to predict new stylistic words at the positions masked by the trained mask predictor G. The third phase is to train a styled filling model by fine-tuning cloze model C pre-trained on the target-style corpus. This fine-tuning process is also supervised by a frozen classifier pre-trained on both source-style and target-style corpus. The network architecture is illustrated in Figure1.

To simplify the description of the model, we only discuss the style transfer process when s_x and s_y are respectively treated as the source and target style with source corpus $\mathcal{D}_x$ and target corpus $\mathcal{D}_y$. A similar process is adopted when we transfer the text from style s_y to s_x.

3.3 Mask Predictor

The mask predictor takes a source text X as the input and produces a mask vector $\mathbf{g} = [g_1, g_2, \cdots, g_n]$, where $g_t \in \{0, 1\}$ implies whether the t^{th} word in text X is a stylistic word. $g_t = 0$ indicates that the t^{th} word in the source text is related to the style of the text and thus should be masked out when transferring to target style. $g_t = 1$ indicates the t^{th} token of the source text is more related to content other than style and therefore should be preserved. For example, if X is "The waiters are *friendly* and *nice*", the mask predictor should generate a mask vector $\mathbf{g} = [1, 1, 1, 0, 1, 0]$, which indicates to mask out the stylistic words *friendly* and *nice*. A masked sequence $\tilde{X}$ is then produced by replacing stylistic words in X with a special token [MASK]. Namely, "The waiters are [MASK] and [MASK]".

In this paper, we propose a novel mask predictor by an adversarial gating strategy that simultaneously trains a Discriminator D to identify the style of the input text and a Gate Generator G to cheat the discriminator and predict the mask vector. We next introduce the Discriminator, Gate Generator, and the adversarial training process in detail.

Gate Generator. The Gate Generator first encodes the input text X by a BERT encoder, and produces a sequence of hidden states $\{\mathbf{h}_t\}$, each corresponds to a word in X. Let $\bar{\mathbf{h}}$ be the average vector of all hidden states in $\{\mathbf{h}_t\}$, the binary indicator g_t for the t^{th} word in X is then computed by

$$g_t = \frac{1}{1 + \exp(-\mathbf{h}_t^{\mathrm{T}} \bar{\mathbf{h}})} \tag{1}$$

In practice, the number of words that should be masked in a text is unknown to the model, and the Gate Generator tends to mask out more words than needed. To overcome this unexpected over-masking problem, we regularize the Gate Generator by the following loss $\mathcal{L}_G$.

$$\mathcal{L}_G = \lambda \sqrt{(\sum_{t=1}^{|X|} g_t - |X|(1-\alpha))^2} \tag{2}$$

where α and λ are hyperparameters.

$\mathcal{L}_G$ makes the number of masked words controllable. It encourages the mask predictor to mask out no more than $\alpha|X|$ tokens, thus preventing the model from masking more words than expected. We can also analyze the trade-off between style transfer strength and content reservation by adjusting α.

Discriminator. The Discriminator D takes the masked word embedding sequence u of a text as input and identifies the style s_u of the text. Specifically, let $\mathbf{x}_t$ be the embedding of the t^{th} word in X, then the t^{th} element in the masked word embedding sequence $\tilde{\mathbf{x}}_t$ is computed by

$$\tilde{\mathbf{x}}_t = g_t \mathbf{x}_t \tag{3}$$

A bidirectional GRU is then used to summarize the masked word embedding sequence to a vector representation $\mathbf{m}$ followed by a softmax classifier

$$P_D = \text{softmax}(\mathbf{W}_D \mathbf{m} + b_D) \tag{4}$$

where P_D is a two-dimensional vector that represents the probabilities of a text belonging to s_x and s_y.

Let s_u be the style of u, the Discriminator D is then trained on the total dataset $\{\mathcal{D}_x, \mathcal{D}_y\}$ with the following cross-entropy loss:

$$\mathcal{L}_D = -\mathbb{E}_{D,u \sim \{\mathcal{D}_x, \mathcal{D}_y\}}[log P_D(s_u|u)] \tag{5}$$

It is worth noting that as the Discriminator D is also used as a frozen classifier to supervise the style transfer model at the third training phase, it is thus firstly pre-trained on the total dataset $\{\mathcal{D}_x, \mathcal{D}_y\}$. It is optional to keep this pre-trained classifier as the Discriminator (Fixed θ^D) or further tune it (Learnable θ^D) in adversarial training.

Adversarial Training. The Gate Generator G and Discriminator D are trained in an adversarial manner. Specifically, the Gate Generator G is trained to generate a mask vector that indicates the positions of the stylistic words and challenges the Discriminator D, whereas the Discriminator D tries to identify the source style. The loss of the adversarial training process is:

$$\mathcal{L}_{adv} = -w_D \mathcal{L}_D + w_G \mathcal{L}_G \tag{6}$$

where w_D and w_G are hyperparameters that control the training of Gate Generator G and Discriminator D. The two components are trained iteratively to enable the Gate Generator to automatically produce mask vectors that indicate the stylistic words. We expect this adversarial gating strategy without the need for additional statistical information and manually set threshold to be a more powerful alternative to existing mask techniques.

3.4 Cloze Model

After obtaining the mask vector of a text, the next training phase is to learn a model that can replace the [MASK] tokens in the masked sequence with stylistic words. This can be realized by introducing a Cloze Model C that completes the words at the positions of the [MASK] tokens based on the contextual information. In this work, we use BERT-based Masked Language Model (Devlin et al., 2018) as the Cloze model C_x, C_y and pre-train them with corpus $\mathcal{D}_x$, $\mathcal{D}_y$ respectively.

To simplify the discussion, we take the pre-training process of C_x as an example, with C_y pre-trained in the same way. The Cloze Model C_x is trained to recover the stylistic words in X from the masked text $\tilde{X}$ at the positions of [MASK] tokens. At each position i of sequence $\tilde{X}$, the Cloze Model C_x outputs a score vector $\mathbf{c}_i = \{c_{i1}, c_{i2}, \cdots, c_{i|V|}\}$, where V denotes the vocabulary. For each position of the [MASK] token, we can then compute its probabilities of being replaced by the words in the vocabulary as

$$P_C^i = \text{softmax}(\mathbf{c}_i) \tag{7}$$

The Cloze Model C_x is also optimized by the cross-entropy loss based on the expected stylistic word in text X.

3.5 Styled Filling

In the second training phase, the Cloze Model C_y is optimized to recover the stylistic words masked by the Mask Predictor, it, however, can not be directly applied to style transferring because it only learns to recover the words from s_y. To enable the Cloze Model C_y to fill in words with s_y, we introduce a third training phase to fine-tune the Cloze Model C_y on the source corpus $\mathcal{D}_x$. This fine-tune phase furthermore takes into account the diversity and semantic consistency of the recovered stylistic words. We next introduce this training phase in detail.

Styled Filling Guided by Discriminator. To use the Cloze Model C_y in filling in target-style words, we utilize the frozen classifier D pre-trained on the total dataset $\{\mathcal{D}_x, \mathcal{D}_y\}$ to supervise the Cloze Model C_y. As this frozen classifier has powerful style classification ability (97.5% and 99.6% accuracy on the validation set of Yelp and IMDB respectively), it provides positive guidance encouraging the Cloze Model C_y to replace stylistic words from s_x with the target style s_y, when the classifier D is forced to identify the replaced text as a target-style text. Specifically, let $\tilde{x}_i$ denotes the i^{th} word in the masked sequence $\tilde{X}$ of target style and V_j denotes the j^{th} word in the vocabulary, we define a function σ upon each output score c_{ij} of the Cloze Model C_y as follows

$$\sigma(c_{ij}) = \begin{cases} c_{ij}, & g_i = 0, \\ 0, & g_i = 1, \tilde{x}_i \neq V_j, \\ 1, & g_i = 1, \tilde{x}_i = V_j, \end{cases} \tag{8}$$

The above function keeps the scores of the stylistic words unchanged and re-scales the scores of non-stylistic words. As the inputs to the frozen classifier D is the word-embedding sequence, and the back-propagation would be failed with hard selection of word embedding, we thus choose to approximate the word embedding by weight sum embeddings in the vocabulary. To that end, we recompute the probabilities of each position on the vocabulary by

$$P_V^i = \sigma(\text{softmax}(\frac{\mathbf{c}_i}{\beta})) \tag{9}$$

where $\beta < 1$ is a hyperparameter that ensures the weighted summed word embedding close to real word embedding by a sharpness operation. Let $\mathbf{v}_j$ as the word embedding of V_j, the i^{th} embedding input to the frozen classifier D is then computed as

$$\mathbf{o}_i = \sum_{j=1}^{|V|} P_V^{ij} \mathbf{v}_j \tag{10}$$

where P_V^{ij} is the j^{th} element of P_V^i. The style of the input text P_S is then predicted by inputting the embedding sequence $\{\mathbf{o}_i\}$ and we have the following cross-entropy loss

$$\mathcal{L}_{tra} = -\mathbb{E}[\log P_S(s_y|\{\mathbf{o}_i\})] \tag{11}$$

Diversity Loss. Supervision from pre-trained classifier D may lead the Cloze Model C_y to repeatedly produce high-frequency stylistic words and result in poor diversity. To cope with this problem, we introduce a diversity loss $\mathcal{L}_{div}$ that can increase the diversity of generated words by maximizing the entropy. Specifically, let $\hat{\mathbf{c}}_i$ denote the score vector of the i^{th} generated word produced by the Cloze Model C_y and P_B^{ij} denote the probability of this word to be the j^{th} vocabulary word, we have

$$P_B^i = \text{softmax}(\hat{\mathbf{c}}_i) \tag{12}$$

Let $\bar{P}_B$ denote the vocabulary distribution averaged over all replaced words and $\bar{P}_B^j$ denote its j^{th} element, the entropy loss is then defined as

$$\mathcal{L}_{div} = \sum_{j=1}^{|V|} \bar{P}_B^j \log \bar{P}_B^j \tag{13}$$

The above entropy loss encourages the generated words to be more evenly distributed in the vocabulary and prevents the Cloze Model from constantly generating high-frequency stylistic words, thus increasing the diversity of generated words.

Consistency Loss. We expect the Cloze Model to maintain semantic consistency with the source-style text while transferring to the target style. For example, when the source-style text is *"The service is fast."*, we prefer the output *"The service is slow."* to *"The service is poor."* We design a consistency loss $\mathcal{L}_{con}$ to keep the semantic consistency of the model. Note that word embedding is constructed based on the assumption that words in similar contexts should have similar meaning (Hill et al., 2014). The key idea is that source stylistic words usually share similar context with their antonyms and thus have a closer word embedding vector compared to other words in the target domain. Specifically, let $\mathbf{E}_x$ denotes the average embedding of the stylistic words in the source text. $\mathbf{E}_y$ and $\hat{P} = \{\hat{p}_k\}$ denote the average embedding of maximum probability filled words with the target style and corresponding probabilities. The consistency loss is then formulated as

$$\mathcal{L}_{con} = (\cos(\mathbf{E}_x, \mathbf{E}_y) - 1) \times \log(\prod_{\hat{P}} \hat{p_k}) \tag{14}$$

where function *cos* denotes the cosine similarity.

The overall loss of style transferring is:

$$\mathcal{L} = w_1\mathcal{L}_{tra} + w_2\mathcal{L}_{div} + w_3\mathcal{L}_{con} \tag{15}$$

where w_1, w_2 and w_3 are hyperparameters that control the three components of loss functions.

4 Experiment

4.1 Dataset

The proposed models are evaluated on Yelp (Shen et al., 2017) and IMDB (Dai et al., 2019) data sets. Yelp consists of review data for businesses including restaurants and home services. Following previous works, we split the Yelp into 444,101 texts for training, 63,483 texts for validation and 126,670 texts for test. As for IMDB movie review data set, it consists of 366,466 texts for training, 4,000 texts for validation and 2,000 texts for test set.

4.2 Evaluation Metrics

To comprehensively evaluate the proposed model, we conduct two aspects of evaluation. Specifically, we evaluate the model on four automatic metrics and human evaluation metric.

Automatic Evaluation Metrics. Automatic evaluation metrics include the follows: **ACC** measures how accurately the model transfer the text style. **BLEU** compares the gold reference and transferred text to measure how well the non-stylistic text tokens are retained after being transferred from the original text (Papineni et al., 2002). **PPL** is used to measure the fluency of the transferred text (Lee et al., 2021; Heafield, 2011). **G-mean** is the geometric mean of BLEU and ACC.

Human Evaluation. We conduct human evaluation to more flexibly and comprehensively evaluate the models. Based on the experimental results on Yelp and IMDB, the style transfer accuracy, content preservation and fluency are measured separately. We randomly sample 100 outputs from test sets for each model. Given the source style and source text, the annotators are asked to score the generated text in the range from 1 (Very Bad) to 5 (Very Good). We compare the models with average scores given by three annotators as shown in Table 2.

Baseline Models. We compare AMSF with the following models: Dis VAE (John et al., 2018), T-VAE-VF (Nangi et al., 2021) , D&R (Li et al., 2018), B-GST (Sudhakar et al., 2019), RACoLN (Lee et al., 2021), NAST (Huang et al., 2021), DGST (Li et al., 2020), ControlledGen (Tian et al., 2018), StyleTransformer (Dai et al., 2019).

4.3 Implementation Details

In the experiment, we set the dimensions of word embedding and hidden state in GRU as 512 and 250, respectively. Following the previous work (Wolf et al., 2020), we train the BERT model with batch size of 128 and 5e-5 lr. When training the mask predictor with a learnable discriminator, we set $w_G = 0.6, w_D = 2$ and the training process runs 45 epochs on Yelp and 50 epochs on IMDB. When the discriminator is fixed, we set $w_G = 1.5, w_D = 0.1$ and the training process runs 20 epochs on Yelp and 30 epochs on IMDB. When training the Cloze model, the pre-training runs 50 epochs with dataset masked by mask predictor. During fine-tuning, Cloze Models runs 10 epochs with masked dataset masked by mask predictor with $w_1 = 1, w_2 = 0.3, w_3 = 1.5, \beta = 0.5$. The hyper-parameters are tuned by experience and the models are trained on V100 NVLINK GPUs.

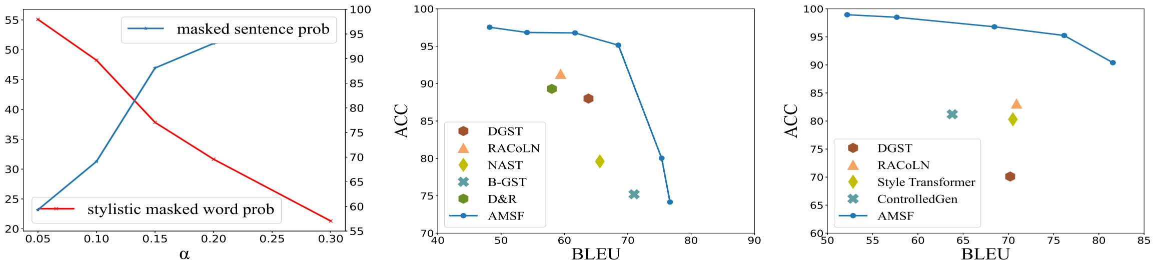
Model On Yelp	**BLEU**	**ACC**	**PPL**	**G-mean**
Dis VAE (John et al., 2018)	47.0	93.0	32.0	66.1
D&R (Li et al., 2018)	58.0	89.3	90.0	72.2
B-GST (Sudhakar et al., 2019)	**71.0**	75.2	38.6	73.1
DGST (Li et al., 2020)	63.8	88.0	–	74.9
T-VAE-CF (Nangi et al., 2021)	34.6	89.9	**15.0**	55.7
RACoLN (Lee et al., 2021)	59.4	91.3	60.1	73.6
NAST (Huang et al., 2021)	65.5	79.6	70.0	72.2
AMSF (Learnable θ^D)	62.4	**98.1**	66.4	78.2
AMSF (Fixed θ^D)	68.5	96.1	62.6	**81.2**
Model On IMDB				
DGST (Li et al., 2020)	70.2	70.1	–	70.1
ContGen (Tian et al., 2018)	63.8	81.2	119.7	71.2
STransformer (Dai et al., 2019)	70.5	80.3	105	75.2
RACoLN (Lee et al., 2021)	70.9	83.1	**45.3**	76.8
AMSF (Learnable θ^D)	75.3	95.2	50.3	84.7
AMSF (Fixed θ^D)	**76.2**	**95.3**	52.1	**85.2**

Table 1: The performance of our model **AMSF** and baseline models on two transfer directions on Yelp and IMDB. θ^D is the parameters of the discriminator D.

4.4 Results and Analysis

Overall Performance On Automatic Evaluation. The Automatic evaluation results on Yelp and IMDB are presented in Table 1 where AMSF (Learnable θ^D) refers to mask predictor trained with discriminator, and AMSF (Fixed θ^D) refers to mask predictor with fixed discriminator.

On both Yelp and IMDB, all of our models outperform the baseline models with G-mean score and AMSF (Fixed θ^D) remarkably improves on G-mean by at least 7 points on Yelp and 8 points on IMDB. In terms of ACC score, AMSF reach



(a) α control (b) trade-off on Yelp (c) trade-off on IMDB

Figure 2: (a) The percentage of stylistic words in masked words (red line, left axis) and ratios of text containing masked words (blue line, right axis) by adjusting α. (b) The trade-off curve between the content preservation and the style accuracy on Yelp. (c) The trade-off curve between the content preservation and the style accuracy on IMDB.

the SOTA score of 98.1% and 95.3% respectively. As for content preservation, our proposed models achieved competitive results. Especially, on IMDB, AMSF achieve the highest BLEU.

It is worth mentioning that there is a tendency of trade-off between style accuracy and content preservation. As a result, measuring the balanced performance of these two metrics, G-mean, which is the geometric mean of ACC and BLEU, is more important. As shown in the Table 1, our models outperform all other baselines on G-mean. In conclusion, automatic evaluation results on Yelp data set and IMDB data set confirm that AMSF-based models are able to attain SOTA score on ACC and at the same time get highly competitive scores on BLEU. These experiment results prove that AMSF can achieve a balanced performance of both style accuracy and content preservation.

Model	Yelp			IMDB		
	Sty	Cont	Flu	Sty	Cont	Flu
Dis VAE	3.41	2.35	3.83	2.85	3.36	3.75
DGST	3.58	4.10	**4.21**	3.52	4.11	4.13
AMSF (Learnable θ^D)	**4.22**	3.93	4.09	4.29	4.31	**4.29**
AMSF (Fixed θ^D)	4.21	**4.35**	4.04	**4.35**	**4.37**	4.23

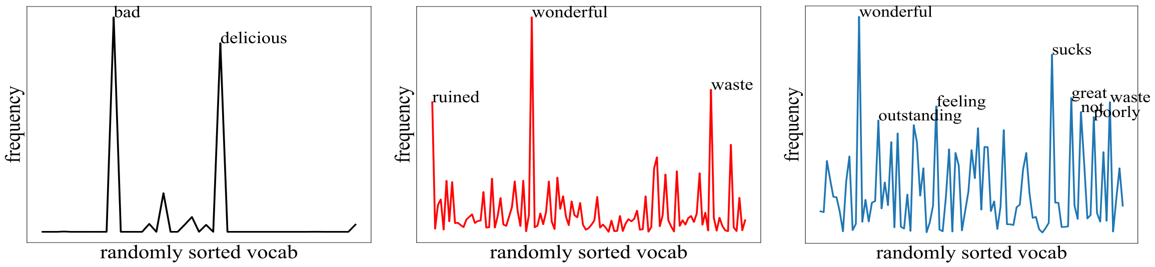
Table 2: Human evaluation result.

α	Yelp			IMDB		
	BLEU	ACC	G-mean	BLEU	ACC	G-mean
0.10	76.67	74.16	75.40	81.55	90.40	85.86
0.15	75.35	80.03	77.65	76.17	95.25	85.18
0.20	68.52	96.14	81.16	68.46	96.80	81.41
0.25	61.68	96.78	77.26	57.67	98.50	75.37
0.30	54.10	96.84	72.38	52.19	98.95	71.86

Table 3: The trade-off between ACC and BLEU by adjusting the ratio α in BERT based Mask Predictor on Yelp and IMDB.

Overall Performance On Human Evaluation. We compared our model AMSF (Learnable θ^D) and AMSF (Fixed θ^D) with Disen VAE and DGST. The average scores evaluated by three annotators on three dimensions: style transfer strength (Sty), content preservation (Cont) and fluency (Flu) are shown in Table 2. Our models not only outperform other models in terms of style transfer strength and content preservation, which is in accordance with the results of the automated evaluation, but their performance on fluency is also competitive.

Trade-off between BLEU and ACC. We further explore and analyze the experiment result of the trade-off between BLEU and ACC. This can be analysed by adjusting a masking ratio of stylistic words. Specifically, the mask predictor's hyperparameter α is used to adjust the ratio of masking stylistic words in the source text. As shown in Figure 2 (a), it can be observed that the likelihood of text being masked is increased and the percentage of stylistic words in all masked words is decreased when α is bigger (for instance, when α is 0.30 rather than 0.05). Therefore, by adjusting α, we can achieve a compromise between style transfer accuracy and content preservation. That is, raising α leads to more replacement in text and hence a lower BLEU score, but on the other hand, it increases transfer accuracy, as shown in Table 3. Figure 2 (b) and (c) are the trade-off curve depicting performance of our model compared with other baseline models (Li et al., 2020; Lee et al., 2021; Huang et al., 2021; Sudhakar et al., 2019; Li et al., 2018) on Yelp and (Li et al., 2020; Lee et al., 2021; Dai et al., 2019; Tian et al., 2018) on IMDB. The score points of the baseline models are below the trade-off curve of our model, explaining that our model achieves higher transfer accuracy when the content preservation score is similar and achieves better content preservation when the style transfer accuracy is at the same level.



(a) without diversity loss

(b) diversity loss weight set 0.3

(c) diversity loss weight set 0.5

Figure 3: (a) The predicted vocab frequency without a diversity loss. (b) The predicted vocab frequency when the weight of diversity loss=0.3. (c) The predicted vocab frequency when the weight=0.5.

Mask Predictor Performance. Following the previous works that use attention scores as an indicator of masking stylistic words, we trained a BERT structural attention-based classifier on Yelp and utilized the average attention score as the threshold for filtering stylistic words. This attention-based classifier obtained 98% accuracy on the validation set. Data is sampled from Yelp with manually labelled stylistic words to evaluate the performance of mask predictors trained in adversarial and attention methods. The precision score and recall score on sampled data are shown in Teble 4.

Our adversarial strategy (Adv BERT) surpasses the attention-based method (Att BERT) in terms of precision, recall and F1-score. The significant improvement in precision score, in particular, demonstrates that our method can precisely mask stylistic words without compromising non-stylistic words.

Method	Precision	Recall	F1
Att BERT	38.6	84.6	53.0
Adv BERT	82.8	92.3	87.3

Table 4: The performance of the adversarial mask predictor and the attention-based mask predictor on Yelp sampled data.

Weight	Trace↑	Entropy↑
0.0	40.03	2.15
0.3	72.22	5.43
0.5	90.04	5.68

Table 5: Trace of covariance matrices of word embedding and entropy over predicted words distribution with different weights of diversity loss.

Effectiveness of diversity loss. To measure the diversity of the generated target-style text, the trace of the covariance matrix of word embedding (Trace) and the entropy of the infilled word distribution (Entropy) on test dataset are introduced. For the Trace, a higher score indicates greater variance in predicted words. For Entropy, a higher value denotes a more diversified distribution. We conduct experiments on Yelp data by varying the weight of L_{div} in training and evaluate the Trace and Entropy metrics. As illustrated in Table 5, it has been shown that more diverse words are predicted with the increasing weight of diversity loss. This fact confirms that our model effectively ensures the diversity of the generated words in target style.

	Source word	With L_{con}	No L_{con}
Yelp	best disappointed fast definitely	worst impressed slow not	worse charismatic annoyed neither
IMDB	regret enjoyed disaster confusing	hope hated masterpiece touching	kidding insulted hooked delicious

Table 6: Stylistic words from the source domain and their transferred words by AMSF with or without L_{con}.

The distributions of the predicted words over vocabulary with different weights of L_{div} are illustrated in Figure3. In Figure 3 (a), it is shown that the model tends to repeatedly predict the same word to cater to the classifier without punishment L_{div}. In Figure 3 (b), it is observed that the predicted vocabularies are more diverse as the weight of diversity loss is increased to 0.3. Figure 3 (c) illustrates the vocabulary distribution when the weight of diversity loss is set to 0.5. It is observed that the occurring vocabs are more evenly distributed in this setting, however, the non-stylistic words such as *feeling* are predicted with higher frequency. This suggests that by adjusting the weight of diversity loss in the training phase, we may be able to keep a balance between linguistic variety and stylistic correctness in the transferred text.

Source	the lobby staff made no effort to accommodate us with a different room .
$\alpha = 0.1$	the lobby staff made **excellent** effort to accommodate us with a different room .
$\alpha = 0.2$	the lobby staff made **excellent** effort to accommodate us with a **wonderful** room .
$\alpha = 0.3$	the lobby staff **did great** effort **always** accommodate us **enjoyed** a **nice** room .
Source	the waitress that helped us was so sweet she explained everything on the menu .
$\alpha = 0.1$	the waitress that helped us was so **rude** she explained everything on the menu .
$\alpha = 0.2$	the waitress that **ignored** us was so **rude** she **ruined** everything on the menu .
$\alpha = 0.3$	the waitress **walked ignored** us was **rude like** she **did not know** the menu .

Table 7: The examples of style transfer on Yelp.

Source	it has lots of humor which does not require much thinking .
$\alpha = 0.1$	it has **lack** of humor which does not require much thinking .
$\alpha = 0.2$	it has **nothing** of humor which does **badly** require much thinking .
$\alpha = 0.3$	it **complete lacks** of humor and does **badly** require much thinking .
Source	but his performance is plain stupid , both with respect the lines uttered and the acting .
$\alpha = 0.1$	but his performance is **very moving** , both with respect the lines uttered and the acting .
$\alpha = 0.2$	but his performance is **very moving** , both with **loved** the lines uttered and the acting .
$\alpha = 0.3$	but his performance is **very well** , both with respect the **subtle humor** and the acting .

Table 8: The examples of style transfer on IMDB.

Effectiveness of consistency loss. The effectiveness of consistency loss is assessed by comparing aligned word pairs generated by the models trained with and without the consistency loss. Some stylistic words from the source text and their most frequently transferred words on Yelp and IMDB are listed in Table6. The result indicates that the model trained with L_{con} performs better in terms of maintaining lexical consistency and transferring to antonyms when transferring the same stylistic word from the source domain to the target domain.

Style Transfer Examples. The examples of transferred text results with different trade-off ratios α corresponding to the source text on two data sets are shown in 7 and 8 with replaced stylistic words emphasized.

5 Conclusion

In this paper, we proposed adversarial masking and styled filling model AMSF to address the problem of text style transfer with unparalleled corpus. AMSF improves the word masking quality by training the mask predictor in an adversarial way and promotes the diversity and semantic consistency of generated sentences by regularization losses. The experimental results on Yelp and IMDB data sets demonstrate that our model is competitive in terms of content consistency and transfer strength. Human evaluation results further corroborate our style transfer model's superiority as well as competitiveness in fluency. The effectiveness of our proposed approach in promoting language diversity and semantic consistency is also verified by the ablation study.

Limitations

It is also worth noting that our model simply substitutes the stylistic words from the source text right in the same place. This pattern is not flexible enough when it comes to more intricate cases. Besides, the masking and filling scheme meets the inherent linguistic properties of the sentiment text style transfer task but is not necessarily applicable to other domains. In future work, we will improve our model and conduct experiments on domains other than sentiment.

Acknowledgement

This work was supported in part by the National Key R&D Program of China under Grant 2021ZD0110700, in part by the Fundamental Research Funds for the Central Universities, in part by the State Key Laboratory of Software Development Environment.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Ning Dai, Jianze Liang, Xipeng Qiu, and Xuanjing Huang. 2019. Style transformer: Unpaired text style transfer without disentangled latent representation. *arXiv preprint arXiv:1905.05621*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Zhenxin Fu, Xiaoye Tan, Nanyun Peng, Dongyan Zhao, and Rui Yan. 2018. Style transfer in text: Exploration and evaluation. In *AAAI*, volume 32.

Kenneth Heafield. 2011. Kenlm: Faster and smaller language model queries. In *Proceedings of the sixth workshop on statistical machine translation*, pages 187–197.

Felix Hill, Kyunghyun Cho, Sebastien Jean, Coline Devin, and Yoshua Bengio. 2014. Embedding word similarity with neural machine translation. *arXiv preprint arXiv:1412.6448*.

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

Fei Huang, Zikai Chen, Chen Henry Wu, Qihan Guo, Xiaoyan Zhu, and Minlie Huang. 2021. Nast: A non-autoregressive generator with word alignment for unsupervised text style transfer. *arXiv preprint arXiv:2106.02210*.

Di Jin, Zhijing Jin, Joey Tianyi Zhou, Lisa Orii, and Peter Szolovits. 2020. Hooks in the headline: Learning to generate headlines with controlled styles. *arXiv preprint arXiv:2004.01980*.

Vineet John, Lili Mou, Hareesh Bahuleyan, and Olga Vechtomova. 2018. Disentangled representation learning for non-parallel text style transfer. *arXiv preprint arXiv:1808.04339*.

Diederik P Kingma and Max Welling. 2013. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.

André Klahold and Madjid Fathi. 2020. *Computer aided writing*. Springer.

Dongkyu Lee, Zhiliang Tian, Lanqing Xue, and Nevin L Zhang. 2021. Enhancing content preservation in text style transfer using reverse attention and conditional layer normalization. *arXiv preprint arXiv:2108.00449*.

Juncen Li, Robin Jia, He He, and Percy Liang. 2018. Delete, retrieve, generate: A simple approach to sentiment and style transfer. *arXiv preprint arXiv:1804.06437*.

Xiao Li, Guanyi Chen, Chenghua Lin, and Ruizhe Li. 2020. Dgst: a dual-generator network for text style transfer. *arXiv preprint arXiv:2010.14557*.

Sharmila Reddy Nangi, Niyati Chhaya, Sopan Khosla, Nikhil Kaushik, and Harshit Nyati. 2021. Counterfactuals to control latent disentangled text representations for style transfer. In *ACL*, pages 40–48.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *ACL*, pages 311–318.

Tianxiao Shen, Tao Lei, Regina Barzilay, and Tommi Jaakkola. 2017. Style transfer from non-parallel text by cross-alignment. *arXiv preprint arXiv:1705.09655*.

Akhilesh Sudhakar, Bhargav Upadhyay, and Arjun Maheswaran. 2019. Transforming delete, retrieve, generate approach for controlled text style transfer. *arXiv preprint arXiv:1908.09368*.

Youzhi Tian, Zhiting Hu, and Zhou Yu. 2018. Structured content preservation for unsupervised text style transfer. *arXiv preprint arXiv:1810.06526*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *NeurIPS*, pages 5998–6008.

Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. 2008. Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th international conference on Machine learning*, pages 1096–1103.

Thomas Wolf, Julien Chaumond, Lysandre Debut, Victor Sanh, Clement Delangue, Anthony Moi, Pierric Cistac, Morgan Funtowicz, Joe Davison, Sam Shleifer, et al. 2020. Transformers: State-of-the-art natural language processing. In *EMNLP*, pages 38–45.

Xing Wu, Tao Zhang, Liangjun Zang, Jizhong Han, and Songlin Hu. 2019. " mask and infill": Applying masked language model to sentiment transfer. *arXiv preprint arXiv:1908.08039*.

Jingjing Xu, Xu Sun, Qi Zeng, Xuancheng Ren, Xiaodong Zhang, Houfeng Wang, and Wenjie Li. 2018. Unpaired sentiment-to-sentiment translation: A cycled reinforcement learning approach. *arXiv preprint arXiv:1805.05181*.