Denoising AutoEncoder Based Delete and Generate Approach for Text Style Transfer

Ting Hu(✉), Haojin Yang, and Christoph Meinel

Hasso Plattner Institute, University of Potsdam, Potsdam, Germany
{ting.hu,haojin.yang,meinel}@hpi.de

Abstract. Text style transfer task is transferring sentences to other styles while preserving the semantics as much as possible. In this work, we study a two-step text style transfer method on non-parallel datasets. In the first step, the style-relevant words are detected and deleted from the sentences in the source style corpus. In the second step, the remaining style-devoid contents are fed into a Natural Language Generation model to produce sentences in the target style. The model consists of a style encoder and a pre-trained DenoisingAutoEncoder. The former extracts style features of each style corpus and the latter reconstructs source sentences during training and generates sentences in the target style during inference from given contents. We conduct experiments on two text sentiment transfer datasets and comprehensive comparisons with other relevant methods in terms of several evaluation aspects. Evaluation results show that our method outperforms others in terms of sentence fluency and achieves a decent tradeoff between content preservation and style transfer intensity. The superior performance on the Caption dataset illustrates our method's potential advantage on occasions of limited data.

Keywords: Text style transfer · Denoising autoencoder · Natural Language Generation

1 Introduction

Text style transfer is a vibrant research area that attracts sustained attention [4,5,9,24]. The task is to transfer given sentences into other styles, meanwhile, preserve the semantics as far as possible [2]. The styles refer to pre-defined categories of texts such as sentiment [7], formality [16], and gender [13]. Text style transfer approaches have been integrated into many practical applications [3]. The most relevant example is widely used writing tools [3], where text style transfer methods enable users to switch their writings among different styles while preserving the contents. Though many algorithms have been explored on parallel text style transfer task, the lack of parallel data leads to more recent research on the non-parallel task setting, where only corpora in different styles are available.

The predominant approaches manage to either disentangle the style features and semantic information in the latent representations of texts or exteriorly disentangle style-relevant words and style-devoid contents [7,11,19]. In general, the

I. Farkaš et al. (Eds.): ICANN 2021, LNCS 12893, pp. 41–52, 2021.
https://doi.org/10.1007/978-3-030-86365-4_4

texts' latent representations are obtained by models like AutoEncoder (AE) [18] and Variational AutoEncoder (VAE) [8]. The disentanglement can be achieved by applying strategies such as adversarial training [20]. However, the latent space's non-smoothness may lead to influent sentences, and the introduction of adversarial training poses a rise in the training instability. In terms of exterior style and contents disentanglement, [7] firstly proposes the three-step text style transfer process, where the style related n-grams are deleted from the sentence, the corresponding n-grams in the target style are then retrieved, and the content remaining in the sentence and retrieved n-grams are combined to generate transfer results. Obviously, the critical factors that impact the transfer performance are how to define the style relevant n-grams, how to retrieve the target-style n-grams, and the generative model's rewriting capability.

Our work follows the idea of [7] while skipping the retrieval step, considering its inflexibility and possible failures. For the first step, we detect and delete style related words or phrases in each source sentence, i.e., style makers, and keep the rest of the sentence, i.e., content. For the second step, we feed the content into a generative model that directly produces a target-style sentence. We think the model's generation capability is important under the circumstance, while the exploration of varied language models in current works [5,19] is insufficient. For instance, [7] employs a Recurrent Neural Network (RNN) as the generative model, [19] uses the pre-trained language model GPT [15], and [5] bases their work on the Transformer [21] architecture. We further regard mapping content texts into intact sentences as a denoising process. The large pre-trained Denoising AutoEncoder (DAE) BART [6] with robust denoising and reasoning capability is therefore employed in our application scenario.

The remaining of the paper is organized as follows. In Sect. 2, we introduce various approaches related to text style transfer. In Sect. 3, we describe our method and the model architecture. In Sect. 4, we describe our experiments and analyze the results. Lastly, we draw the conclusion in Sect. 5. We summarize our contributions as follows.

- We propose a pre-trained Denoising AutoEncoder based framework for the delete and generate approach on the text style transfer task.
- We conduct experiments on Yelp and Caption datasets [7] and detailed comparisons with other variants of three-step text style transfer methods.
- Experiments demonstrate that our approach achieves decent and stable performance on two datasets on different evaluation aspects. The good performance on the Caption dataset indicates its underlying advantage over others in applications where only a small amounts of data are available.

2 Related Work

Text style transfer tasks can be categorized as parallel, non-parallel, and label-free. In parallel data setting, pairs of sentences with different styles are provided. The seq-to-seq model and its variants [16] are commonly used in this task. Since parallel text pairs could be difficult to collect, many methods [5,7,24] focus on

the non-parallel setting, where only the source and the target style corpus are available. Some recent works [4,9,17] explore label-free approaches that further get rid of any training style labels and manipulate sentences into arbitrary styles during inference. In this work, we study the non-parallel text style transfer task.

There are three primary methods for text style transfer with non-parallel data: representation disentanglement, back-translation, and sentence editing. Representation disentanglement approaches generally follow the process of encoding, manipulating, and decoding [2]. AE [18], VAE [8], and Generative Adversarial Network [25] have been used to encode the input texts into representations and decode the manipulated representations into target style texts. The manipulation procedure is based on representation disentanglement, where the style information and style-devoid semantic information are disentangled by applying an additional style classifier [9], or adversarial learning [20].

Back-translation is commonly used in machine translation. When it comes to text style transfer, one route is iterative back-translation [24], consisting of two steps: (1) Initialize two specific style transfer models and produce pseudo-parallel corpora. (2) Iteratively update transfer models and produce better pseudo-parallel data. [14] applies online back-translation, where the latent codes devoid of style information are obtained through back-translation, and multiple decoders are then employed to produce texts in different styles.

Our approach belongs to the sentence editing category, firstly proposed by [7]. Unlike representation disentanglement, this type of method directly disentangles style words and content words at the sentence level. The sentence editing process is: (1) Delete the source style markers in each sentence, and obtain the remaining content. (2) Retrieve the counterpart style markers in the target style corpus. (3) Combine the content and the retrieved target markers and generate a fluent transferred sentence.

For the first step, there are multiple ways to detect the markers. [7] defines the salience of an n-gram with respect to the source style to be its relative frequency in the source corpus versus the target corpus. The n-grams of which the salience are higher than a specified threshold are declared as the style markers. [11] calculates the ratio of mean Term Frequency-Inverse Document Frequency (TF-IDF) between two style corpus for each n-gram and regards the normalized ratio as the salience. [19] trains a BERT classifier using the training corpora and considers words with attention weights larger than average as the markers. [22] employs the frequency-ratio method to predict the markers, supplemented by the attention weights method.

The second step is to retrieve each content-only source sentence's closest neighbor in the content-only target corpus. Then the deleted markers in the retrieved content-only target sentence are the corresponding target-style markers. In order to search for the nearest neighbor, sentence embeddings are generally used to evaluate the similarities between the content-only sentences, such as TF-IDF, Glove Embeddings, and Universal Sentence Encoder.

For the generation step, [7] feeds the content and retrieved target markers into an RNN to produce transferred outputs. [19] further feeds them into a

pre-trained language model GPT [15] to generate fluent sentences. Considering markers retrieval may fail if the target corpus does not have sentences similar to that in the source corpus, some works circumvent the retrieval step and train a generative model that directly maps the input contents into sentences in a specific style. [19] concatenates the source style label, the content, and the original complete sentence as the input of GPT and fine-tune it to reconstruct the original sentence. During inference, the model produces the transferred sentence given the target style label concatenated with the content. [5] employs the Transformer [21] architecture, where the encoder's input is the content tokens, and the decoder's input is the original sentences prepended with the style labels. The model is trained to minimize the reconstruction loss and the style loss measured by a pre-trained classifier.

3 Approach

The Non-parallel text style transfer setting is as follows. Source style corpus and target style corpus C_s and C_t are given. Here, we consider negative and positive sentiments as two styles. The sentiment transfer model is supposed to transfer each sentence in C_s to the target sentiment implied in C_t and vice versa. Transferred texts are considered to preserve the semantics of the source sentence and accord with the target style. We divide the text style transfer process into deleting and generating procedures described below.

3.1 Delete

We need to at first detect the markers in both style corpora. Since we are focusing on semantic text transfer in this work, the assumption is that different sentiments can be captured by the most frequent N-grams. As [7] described, the salience of an n-gram u with respect to the source style is defined by its relative frequency in the source corpus C_s, that is,

$$s(u, C_s) = \frac{count(u, C_s) + \lambda}{count(u, C_s) + count(u, C_t) + \lambda} \tag{1}$$

where $count(u, C_s)$ is the number of times that u appears in C_s, $count(u, C_t)$ is the number of times that u appears in C_t, and λ is a smoothing parameter. The n-grams of which the salience scores are above a pre-defined threshold are then regarded as markers. The selection of the threshold is obviously of importance. A lower threshold leads to more markers and fewer contents, which provides a broader space for exploring transfer intensity. A higher threshold eventually results in a stronger content constraint and limited transfer intensity. In effect, this is the tradeoff between content preservation and transfer intensity that many text style transfer models go through.

We also attempted to employ the method of [19] to detect markers, where a BERT classifier pre-trained by two corpora is used to measure words' salience according to their attention weights. However, the detection results are inferior

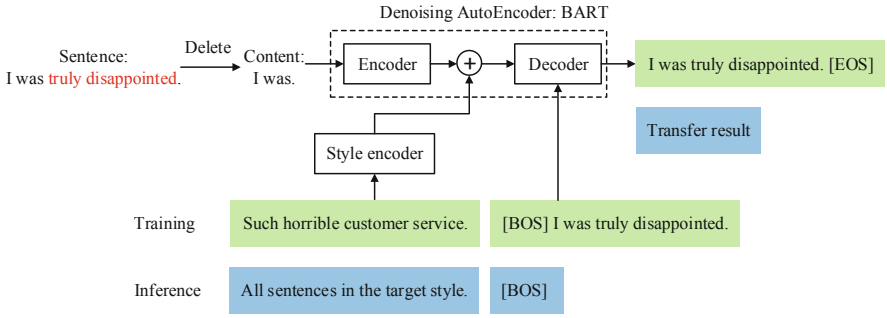

Fig. 1. The two-step text style transfer process. The words in red are deleted markers. The style encoder has the same architecture as the encoder and is initialized by its parameters. Token [BOS] and [EOS] stand for the beginning and the end of a sentence, respectively. When training, a random source-style sentence is fed into the style encoder, and the decoder is trained to reconstruct the original sentence. When inference, all target-style sentences are fed into the style encoder to obtain the mean style vector, based on which the decoder auto-regressively generates the transferred sentence.

to this simple statistical method. The possible reason could be the indirect and unclear relationship between the words' style representation capabilities and their attention weights.

3.2 Generate

For this step, the pre-trained Denoising AutoEncoder BART [6] is used as our basic framework. BART consists of a bidirectional encoder like BERT and a unidirectional decoder like GPT. BART is pre-trained to reconstruct input sentences corrupted by shuffling, tokens deletion, and spans of texts masking. Our basic idea is to regard the marker deleting procedure as the noise that BART is pre-trained to overcome. Hence, the encoder's inputs are the style devoid contents obtained from the deleting step, and the decoder's inputs are the corresponding intact sentences. The encoder-decoder architecture is fine-tuned to reconstruct the original sentences.

However, we need two separate style transfer models for each style under this setting, making style transfer inflexible. Inspired by [4,17], we keep the single BART architecture and employ an additional style encoder to extract style information from sentences in the same style as the original intact sentences. The style encoder has the same architecture as the encoder of BART and is initialized by its parameters. We conduct a max-pooling operation on the final output hidden states of the style encoder to obtain the style vector. Then the style vector is added to the content encoder's top hidden states which are used as the initial hidden states of the decoder.

The complete process framework is illustrated in Fig. 1. During training, the decoder combines the style features extracted by the style encoder and the

Table 1. Dataset statistics.

Dataset	Sentiment	Training set	Dev set	Test set
Yelp	Positive	270k	2000	500
	Negative	180k	2000	500
Caption	Humorous	6000	300	0
	Romantic	6000	300	0
	Factual	0	0	300

content information extracted by the content encoder and reconstructs original sentences. For inference, we only alter the style encoder's input to be a random target style sentence, and the decoder is expected to generate sentences in the target style. In practical implementation, we find that the randomly chosen sentence may lead to weak transfer intensity, considering the dataset is noisy, and sentences have varied style intensity. Consequently, we use the trained style encoder to obtain all style vectors from the source style corpus and the target style corpus, respectively, and take the mean vector on each corpus as the representative style vector, which is then used by the decoder to produce transferred results. The potential benefits of our training strategy are: i) the additional style encoder has seen sentences in both styles and learns to extract the most salient features regarding two styles; ii) even sentences in the same style could have variations in style and semantics, the decoder learns to discard disturbing information and make better use of useful information and becomes more robust.

Our method is different from [17], where only a small set of most representative sentences are selected and contribute to two mean style vectors. The deviation of them is considered as the basic transfer direction, with a hyperparameter β involved in the instance-wise transfer direction determination. In contrast, our model has seen sentences in both styles during training. The style encoder learns to extract the most salient features regarding two styles, and the deviation operation is not a must.

4 Experiments

4.1 Datasets

We conduct experiments on two sentiment style transfer datasets provided by [7]: Yelp and Caption. Yelp dataset contains business reviews on Yelp, and each review is labeled as positive or negative sentiment. The Caption is originally a parallel dataset, where caption pairs in the training set are labeled as romantic and humorous, respectively. Only factual captions are provided in the test set, and the style transfer model is supposed to transfer them into romantic and humorous sentiment. In our implementation, we treat Caption as a non-parallel dataset. The statistics of these datasets are displayed in Table 1, the same as [7]. All models included in our comparison are evaluated on the same test set.

4.2 Models for Comparisons

We compare with three conventional methods related to adversarial learning, including CrossAligned (CA) [18], StyleEmbedding (SE)[1], and MultiDecoder (MD) [1], and method BackTranslation (BT) [14]. Other approaches related to the three-step text style transfer process are the focus of our comparison. DelOnly and DelAndRet are from [7], where the former directly produces the output from the content and the target style with an RNN, and the latter generates the output from the content and retrieved target style markers with an RNN. [19] proposes Generative Style Transformer (GST): B-GST generates the output given the content and the target style, blind to specific markers, and G-GST produces the output given the content and retrieved target style markers. [5] proposes the Stable Style Transformer (SST), which produces outputs from the contents and the style classifier's feedback based on the encoder-decoder framework.

4.3 Evaluation Metrics

Following [2,5,12], we mainly evaluate style transfer results from four perspectives: content preservation, style transfer intensity, fluency, and the similarity to human references.

We employ self-BLEU (s-BLEU) and masked Word Mover Distance (mWMD) [12] to measure content preservation. S-BLEU is the BLEU score between the original sentences and the transferred candidates. To compute mWMD, we mask the style-related words in both the candidates and the references and calculate the minimum distance between the word embeddings of them. A smaller mWMD indicates better content preservation capability.

Accuracy (Acc) and Earth Mover's Distance (EMD) [12] are used to measure the style transfer intensity. We train a BERT classifier using given training data to predict the transferred sentences' style conformity. However, the accuracy provided by the binary classifier could not reveal transfer sentences falling in between two styles. The EMD between the style distributions of the source style corpus and the transfer outputs could better reflect the nuanced style difference. A larger EMD indicates a stronger transfer intensity.

Considering models usually experience the tradeoff between the aspects of content preservation and transfer intensity, we take the Geometric mean Score(GScore) of the aforementioned four metrics: s-BLEU, 1/mWMD, Acc, and EMD, to evaluate the tradeoff capability of different models. We take the inverse of mWMD to make the GScore decrease monotonically with respect to the worse performance.

In terms of fluency, we use general-PPL (g-PPL) and data-PPL (d-PPL) [5] to perform evaluation. G-PPL is obtained by using a pre-trained GPT-2 model and measures how transferred sentences are fluent in terms of massive natural texts used to pre-train GPT-2. D-PPL is obtained by fine-tuning a pre-trained GPT model on training data and measures how transferred results fit the data

distribution of specific style corpora. We then take the GScore of them and use it as the indicator of transferred sentences' fluency.

Since human written references are provided in these datasets, we evaluate the transfer outputs' similarity to human references by BERTScore [23]. Unlike BLEU, which measures the n-gram overlapping between the references and the candidates, BERTScore measures the semantic similarity between them and is demonstrated to have better correlation to human judgments.

4.4 Experiment Details

For marker detecting, n-grams that span up to 4 words are considered as potential style markers, and the smoothing parameter is set to 1. Following the setting of [7], the thresholds of defining style markers are 15 and 5 for Yelp and Caption, accordingly. We use the Pytorch implementation of Transformer by Huggingface[1] for experiments. Our framework's components are built from the base-size BART, consisting of a six-layer encoder and a six-layer decoder with a hidden size of 768. The maximum input sequence length is 60. The framework is fine-tuned up to 10 epochs using cross entropy loss, with a batch size of 32 and a learning rate of 1e-5. The model performs best on the dev set is saved and used for inference on the test set. During inference, we merely conduct greedy search without any results selection module. For evaluation metrics, we use the tool provided at the link[2] to compute mWMD and EMD. Other metrics are from the link[3].

4.5 Result Analysis

The evaluation results on Yelp dataset are listed in Table 2. H:DRG [7] and H:DualRL [10] are taken as references. It is worth noting that two human references do not achieve the best performance in content preservation and style transfer intensity metrics. For instance, Human:DualRL obtains an accuracy of 77% and a self-BLEU score of 37.79, far behind other neural network methods. The possible reasons are that human's definitions of different sentiments are varied, and humans are better at creative sentence rewriting. On the other hand, this reveals the limitation of current widely used evaluation metrics. Automatic evaluation metrics that correlate better to human judgments are to be studied.

As we can see, B-GST [19] attains the best tradeoff between content preservation and style transfer intensity and the highest semantic similarity to human references. Even though GST methods surpass ours, we achieve decent results in the above aspects and the lowest PPL compared with other variants of three-step approaches, demonstrating the transfer results of our methods are more fluent and fit to training texts distribution. We attribute this to the denoising and generation capability of the pre-trained Denoising AutoEncoder we use, considering SST employs the encoder-decoder architecture as well while inferior to ours.

[1] https://github.com/huggingface/transformers.
[2] https://github.com/SenZHANG-GitHub/graph-text-style-transfer.
[3] https://github.com/rungjoo/Stable-Style-Transformer.

Table 2. Automatic evaluation results on Yelp dataset. The best result among methods based on the delete-retrieve-generate approach on each evaluation metric is shown in bold. *GScore* is the Geometric mean of the evaluation results on specific aspects. *Sem* is the semantic similarity between the transfer results and the references.

Model	Content and Style					Fluency			Sem↑
	s-BLEU↑	mWMD↓	Acc↑	EMD↑	GScore↑	d-PPL↓	g-PPL↓	GScore↓	
H:DRG [7]	26.97	0.503	72.8	0.726	7.30	121.2	153.5	136.4	95.83
H:DualRL [10]	37.79	0.388	77.0	0.766	8.71	178.6	196.2	187.2	95.83
CA [18]	17.02	0.512	74.8	0.713	6.49	69.1	319.1	148.5	88.12
SE [1]	71.80	0.880	8.9	0.412	4.16	121.7	379.8	215.0	90.56
MD [1]	40.81	0.580	46.4	0.634	6.75	201.6	642.1	359.8	88.35
BT [14]	0.67	0.757	96.2	0.912	2.97	148.8	67.4	100.1	87.36
DelOnly [7]	33.94	0.454	84.8	0.830	8.52	171.7	279.6	359.8	89.28
DelAndRet [7]	34.48	0.461	**87.7**	**0.855**	8.65	137.0	343.8	219.1	89.39
B-GST [19]	43.45	**0.237**	86.1	0.832	**10.71**	165.6	**184.0**	174.6	**91.78**
G-GST [19]	43.94	0.246	77.2	0.740	10.05	441.4	274.3	348.0	91.15
SST [5]	**49.09**	0.277	70.4	0.661	9.53	197.8	295.9	241.9	90.65
Ours	45.87	0.342	87.2	0.843	9.96	**132.5**	224.7	**172.5**	90.26

According to the results on Caption dataset in Table 3, our approach achieves the best performance regarding the content and style tradeoff and fluency. In our method, the auxiliary style encoder sees sentences in both sentiments and is trained to extract these two sentiments' representative features, which are then fed into the decoder for reconstruction. Under the circumstance of limited data, more salient features are extracted in our approach and results in a stronger transfer intensity and a lower PPL score. In addition, method DelAndRet attains a remarkably high accuracy of 94.7%. Intuitively, the reason is that target style markers are easily and accurately retrieved in parallel sentence pairs, which further leads to the strong transfer intensity.

Two groups of transferred sentences from other three-step relevant methods are displayed in Table 4, where markers are shown in bold. For the first group, method DelAndRet apparently fails to convert the source sentence into the target style, and SST produces a contradictory sentence with both *good* and *wrong* involved. In the second group, the results from DeleteOnly and G-GST are influent with repeated word *love*, and others manage to indicate the romantic sentiment through word *lover* or *loving*. These exactly demonstrate how style transfer methods perform diversely on different instances, making the applications of automatic evaluation metrics on the corpus level indispensable. The overall transfer results of our methods are shared at the link[4].

In conclusion, these variants of the three-step text style transfer method have different strengths and weaknesses. GST methods consistently attain the highest semantic similarity to human references, though human references do not

[4] https://drive.google.com/drive/folders/1H5Jg7psMRpGMWbBXnk5E1WMhq1zvJgIy?usp=sharing.

Table 3. Automatic evaluation results on Caption dataset. The best result among methods based on the delete-retrieve-generate approach on each evaluation metric is shown in bold. *GScore* is the Geometric mean of the evaluation results on specific aspects. *Sem* is the semantic similarity between the transfer results and the references.

Model	Content and Style					Fluency			Sem↑
	s-BLEU↑	mWMD↓	Acc↑	EMD↑	GScore↑	d-PPL↓	g-PPL↓	GScore↓	
Human [7]	16.38	0.345	74.5	0.223	4.06	145.2	144.7	144.9	100.0
CA [18]	0.76	0.485	78.0	0.263	2.38	11.4	75.9	29.4	88.45
SE [1]	30.79	0.226	53.5	0.029	3.81	122.9	404.7	223.0	88.70
MD [1]	22.72	0.247	68.3	0.134	5.39	60.9	239.3	120.7	88.67
DelOnly [7]	39.88	**0.216**	77.3	0.176	7.08	464.3	345.1	400.3	89.38
DelAndRet [7]	33.32	0.243	**94.7**	0.135	6.47	559.9	160.1	299.4	89.51
B-GST [19]	**64.71**	0.345	59.1	0.223	7.05	126.1	140.3	133.0	**90.70**
G-GST [19]	51.43	0.243	57.3	0.135	6.36	1300.5	133.7	417.0	86.40
Ours	45.22	0.247	63.8	**0.228**	**7.18**	**88.2**	**126.9**	**105.8**	90.09

Table 4. Transfer results from different methods on two datasets. For Yelp dataset, the tokens in bold are style makers. Since the given sentence is factual in the test set of Caption, there is no style markers to be deleted in the source.

Yelp: negative → positive	
Source	We sit down and we got some really **slow** and **lazy** service
DelOnly	We sit down and we got some **great** and **quick** service
DelAndRet	We got very nice place to sit down and we got some service
B-GST	We sit and we got some really **good** and **friendly** service
G-GST	We sit and we got some really **amazing** food and **great** service
SST	We sit and we got some really **good** and **wrong** customer service
Ours	We sit down and we got some really **nice** and **fast** service
Caption: factual → romantic	
Source	A brown dog runs with a toy in its mouth
DelOnly	**People in love carrying** a brown dog runs with a toy in its mouth
DelAndRet	A brown dog runs with a toy in its mouth **to meet its lover**
B-GST	A brown dog runs with a toy in it's mouth, **towards his lover**
G-GST	**People in love carrying** a brown dog runs with a toy in its mouth
Ours	A brown dog runs with a toy in its mouth **towards his loving master**

achieve the best performance on these automatic evaluation metrics. DelOnly and DelAndRet perform stably on two datasets while their content preservation capability and style transfer intensity are limited. Ours surpasses the counterpart SST and achieves the best sentence fluency and a decent content and style trade-off. Moreover, our method is advantageous in the case of limited data regarding its good performance on the Caption dataset.

5 Conclusion

We have studied a two-stage exterior style and content disentanglement method for text style transfer. In the first stage, the style markers are detected and deleted from the sentences. In the second stage, the style-irrelevant contents are fed into a generative model to produce sentences in the target style. The model we propose consists of a content encoder, a decoder, and a style encoder. The former two components are directly built from the pre-trained Denoising AutoEncoder BART, and the latter has the same architecture as the content encoder while functioning differently. We conduct experiments on two text sentiment transfer datasets and carry out comprehensive evaluations on the variants of three-step transfer approaches. We hope these can promote a better understanding of the strengths and weaknesses of diverse methods. Moreover, our method's transfer results achieve decent performance regarding content preservation, transfer intensity, and semantics and stand out in terms of sentence fluency. For future work, the study of other marker detecting methods is desirable, considering the current commonly used methods are limited and the detecting results have a significant impact on the following generating step.

References

1. Fu, Z., Tan, X., Peng, N., Zhao, D., Yan, R.: Style transfer in text: exploration and evaluation. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 32 (2018)
2. Jin, D., Jin, Z., Hu, Z., Vechtomova, O., Mihalcea, R.: Deep learning for text style transfer: a survey. arXiv preprint arXiv:2011.00416 (2020)
3. Klahold, A., Fathi, M.: Word processing as writing support. In: Klahold, A., Fathi, M., et al. (eds.) Computer Aided Writing, pp. 21–29. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-27439-9_4
4. Lample, G., Subramanian, S., Smith, E., Denoyer, L., Ranzato, M., Boureau, Y.L.: Multiple-attribute text rewriting. In: International Conference on Learning Representations (2018)
5. Lee, J.: Stable style transformer: delete and generate approach with encoder-decoder for text style transfer. In: Proceedings of the 13th International Conference on Natural Language Generation, pp. 195–204 (2020)
6. Lewis, M., et al.: BART: denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. arXiv preprint arXiv:1910.13461 (2019)
7. Li, J., Jia, R., He, H., Liang, P.: Delete, retrieve, generate: a simple approach to sentiment and style transfer. In: 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL HLT 2018, pp. 1865–1874. Association for Computational Linguistics (ACL) (2018)
8. Liao, Y., Bing, L., Li, P., Shi, S., Lam, W., Zhang, T.: Quase: sequence editing under quantifiable guidance. In: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pp. 3855–3864 (2018)

9. Liu, D., Fu, J., Zhang, Y., Pal, C., Lv, J.: Revision in continuous space: unsupervised text style transfer without adversarial learning. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, pp. 8376–8383 (2020)
10. Luo, F., et al.: A dual reinforcement learning framework for unsupervised text style transfer. arXiv preprint arXiv:1905.10060 (2019)
11. Madaan, A., et al.: Politeness transfer: a tag and generate approach. arXiv preprint arXiv:2004.14257 (2020)
12. Mir, R., Felbo, B., Obradovich, N., Rahwan, I.: Evaluating style transfer for text. arXiv preprint arXiv:1904.02295 (2019)
13. Prabhumoye, S., Chandu, K.R., Salakhutdinov, R., Black, A.W.: "My way of telling a story": persona based grounded story generation. arXiv preprint arXiv:1906.06401 (2019)
14. Prabhumoye, S., Tsvetkov, Y., Salakhutdinov, R., Black, A.W.: Style transfer through back-translation. In: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 866–876 (2018)
15. Radford, A., Narasimhan, K., Salimans, T., Sutskever, I.: Improving language understanding by generative pre-training (2018). https://s3-us-west-2.amazonaws.com/openai-assets/research-covers/language-unsupervised/language_understanding_paper.pdf
16. Rao, S., Tetreault, J.: Dear sir or madam, may i introduce the GYAFC dataset: corpus, benchmarks and metrics for formality style transfer. arXiv preprint arXiv:1803.06535 (2018)
17. Riley, P., Constant, N., Guo, M., Kumar, G., Uthus, D., Parekh, Z.: TextSETTR: label-free text style extraction and tunable targeted restyling. arXiv preprint arXiv:2010.03802 (2020)
18. Shen, T., Lei, T., Barzilay, R., Jaakkola, T.: Style transfer from non-parallel text by cross-alignment. arXiv preprint arXiv:1705.09655 (2017)
19. Sudhakar, A., Upadhyay, B., Maheswaran, A.: Transforming delete, retrieve, generate approach for controlled text style transfer. In: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pp. 3260–3270 (2019)
20. Tian, Y., Hu, Z., Yu, Z.: Structured content preservation for unsupervised text style transfer. arXiv preprint arXiv:1810.06526 (2018)
21. Vaswani, A., et al.: Attention is all you need. arXiv preprint arXiv:1706.03762 (2017)
22. Wu, X., Zhang, T., Zang, L., Han, J., Hu, S.: "Mask and infill": applying masked language model to sentiment transfer. arXiv preprint arXiv:1908.08039 (2019)
23. Zhang, T., Kishore, V., Wu, F., Weinberger, K.Q., Artzi, Y.: BERTScore: evaluating text generation with BERT. arXiv preprint arXiv:1904.09675 (2019)
24. Zhang, Z., et al.: Style transfer as unsupervised machine translation. arXiv preprint arXiv:1808.07894 (2018)
25. Zhao, J., Kim, Y., Zhang, K., Rush, A., LeCun, Y.: Adversarially regularized autoencoders. In: International Conference on Machine Learning, pp. 5902–5911. PMLR (2018)