Improving Neural Text Style Transfer by Introducing Loss Function Sequentiality

Chinmay Rane[1,2], Gaël Dias[1], Alexis Lechervy[1], Asif Ekbal[3]
[1]Normandie Univ, UNICAEN, ENSICAEN, CNRS, GREYC, 14000 Caen, France
[2]ABV-Indian Institute of Information Technology and Management Gwalior, India
[3]Indian Institute of Technology - Patna, India
chinmayrane16@gmail.com,{gael.dias,alexis.lechervy}@unicaen.fr,asif@iitp.ac.in

ABSTRACT

Text style transfer is an important issue for conversational agents as it may adapt utterance production to specific dialogue situations. It consists in introducing a given style within a sentence while preserving its semantics. Within this scope, different strategies have been proposed that either rely on parallel data or take advantage of non-supervised techniques. In this paper, we follow the latter approach and show that the sequential introduction of different loss functions into the learning process can boost the performance of a standard model. We also evidence that combining different style classifiers that either focus on global or local textual information improves sentence generation. Experiments on the Yelp dataset show that our methodology strongly competes with the current state-of-the-art models across style accuracy, grammatical correctness, and content preservation.

CCS CONCEPTS

• **Computing methodologies** → **Natural language generation**;
• **Information systems** → **Sentiment analysis**.

KEYWORDS

Text style transfer, sentiment transfer, loss function sequentiality, combining style classifiers

ACM Reference Format:
Chinmay Rane[1,2], Gaël Dias[1], Alexis Lechervy[1], Asif Ekbal[3]. 2021. Improving Neural Text Style Transfer by Introducing Loss Function Sequentiality. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '21), July 11–15, 2021, Virtual Event, Canada.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3404835.3463026

1 INTRODUCTION

Text style transfer (TST) consists in converting the style of a given sentence into a target style while preserving its content (style-independent) information. This task can linguistically be defined as transforming the connotation of a given message while maintaining its denotation [18]. In practice, TST reflects the ability of language generation systems to produce novel sentences with different diversity levels. As such, TST has a myriad of facets, including sentiment, offensive, and formality transfer[1] that can support conversational agents in dealing with specific dialogue situations [25].

Preliminary works have employed sequence-to-sequence models [1, 11] on parallel corpora and reported to have achieved remarkable performance. However, further research in this direction is limited, mainly due to two reasons: the non-availability of adequate parallel data and the time-consuming process to annotate such datasets.

As a result, there has been a recent surge of interest to perform TST in unsupervised settings [9, 14, 15, 17, 23]. For that purpose, encoder-decoder architectures are proposed that combine different loss functions: mainly transfer style and reconstruction loss functions. Despite impressive results, many works still struggle to retain the semantic information in the translated text while correctly transferring the style information [4].

In this paper, we propose a new research direction inspired by curriculum learning [2]. We hypothesize that the different loss functions of the unsupervised methods should sequentially be introduced in the learning procedure to take into account the different cognitive steps of the generation process. We also propose to combine different style classifiers that either focus on global or local textual information. We speculate that incorporating syntactic (local) and semantic (global) information can positively support style transfer and consequently improve overall generation performance.

To account for our two main contributions, we implement a non-contextual CAST-inspired architecture [4] and test our hypotheses over the Yelp dataset [23] for the task of sentiment transfer. Overall results show that our methodology strongly competes with the current state-of-the-art models, evidencing best results in terms of globalized metrics, namely G2, G2_H4, GM4, and GM4_H4.

2 RELATED WORK

Recent works on TST can broadly be classified into two categories: supervised and unsupervised approaches. Supervised methods assume the presence of parallel corpora and generally make use of sequence-to-sequence (seq2seq) models. Jhamtani et al. [11] were the first to train a seq2seq architecture to convert a text in modern English to Shakespearean English. Later, Carlson et al. [3] trained an attention-based seq2seq model for the Bible prose style transfer.

Unsupervised learning techniques have been pervasively studied because they eliminate the need for parallel corpora. Within this context, some approaches [27, 28, 32] follow an explicit style-content disentanglement strategy, where text from the original style

[1]To name but a few.

SIGIR '21, July 11–15, 2021, Virtual Event, Canada

ACM ISBN 978-1-4503-8037-9/21/07…$15.00
https://doi.org/10.1145/3404835.3463026

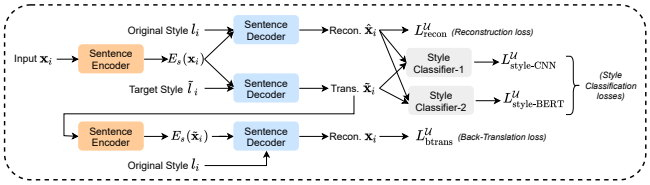

Figure 1: CAST-inspired architecture for non-parallel data. Sentence encoder and decoder are shared across training.

is explicitly replaced to generate text of a target style. For example, the work reported in [15] proposed a *Delete-Retrieve-Generate* approach, which deletes phrases associated with the original style, retrieves new phrases linked to the target style, and uses a neural model to combine them into a final output. Other works adopt an implicit style-content disentanglement methodology that aims at learning the content and style latent representations of a given text [6, 9, 33]. For instance, Shen et al. [23] assumed shared latent content representation across different corpora and trained an auto-encoder with adversarial discriminator to separate content and style information. A similar study [31] demonstrated that using language models instead of style classifiers as discriminators improved the quality of text generation. The back-translation based strategy inspired from the unsupervised machine translation has been used to ensure semantic consistency in the translated text. One of the recent works reported in [4] considered the context on the top of the sentence and proposed a context-aware style transfer (CAST) architecture, where a coherence classifier ensures that the translated sentence is contextually consistent. Some other works have exploited the advantages of reinforcement learning [17, 29] and achieved the state-of-the-art performance over the Yelp dataset [8]. Finally, other studies [14, 30] suggest that the content-style disentanglement is unnecessary for TST. For instance, Riley et al. [22] showed that a robust pre-trained text-to-text model can be adapted to extract a style vector from arbitrary text, which can condition the decoder to perform style transfer. The interested reader can explore more about TST in a complete survey by Hu et al. [8].

In this paper, we propose the use of loss sequentiality to improve sentence generation, and explore the effect of combining local and global latent features to improve style and content disentanglement. As far as we know, this is the first attempt in this direction.

3 METHODOLOGY

3.1 CAST-inspired architecture

Figure 1 illustrates our non-contextual CAST-inspired architecture for non-parallel data. It mainly follows the original architecture presented in [4], but combines two style classifiers to account for style transfer accuracy.

Formally, a style-labelled non-parallel corpus can be represented as $\mathcal{U} = \{(\mathbf{x}_i, l_i)\}_{i=1}^{N}$, where $\mathbf{x}_i$ is the i-th sentence with style l_i. Our TST model consists of a sentence encoder E_s and a sentence decoder D. E_s aims at extracting the semantic representation of the input sentence, which is further fed into D along with the desired style representation to generate some sentence. The sentence representation inferred by the encoder $E_s(\mathbf{x}_i)$ is either concatenated with the target style and passed through D to produce the translated sentence $\tilde{x}_i$, or concatenated with the original style to reconstruct the original sentence $\hat{x}_i$ through D.

Three different losses are introduced to guide the architecture towards style translation: (1) *reconstruction loss*, (2) *back-translation loss*, and (3) *style classification loss*, which are summed up as in equation 1 to give rise to the final loss function.

$$L_{\text{final}} = L^{\mathcal{U}}_{\text{recons}} + L^{\mathcal{U}}_{\text{btrans}} + L^{\mathcal{U}}_{\text{style}} \tag{1}$$

3.1.1 Reconstruction loss. It enforces the neural architecture to accurately recover the original stylistic properties which are lost in the encoded representation, while generating the text. The reconstructed text is denoted as $\hat{\mathbf{x}}_i = D(E_s(\mathbf{x}_i), l_i)$. The reconstruction loss is the categorical cross entropy loss defined in equation 2.

$$L^{\mathcal{U}}_{recon} = - \mathop{\mathbb{E}}_{\mathbf{x}_i \sim \mathcal{U}} \log p_D\left(\mathbf{x}_i \mid E_s\left(\mathbf{x}_i\right), l_i\right) \tag{2}$$

3.1.2 Style classification loss. This loss function assesses whether the generated text contains the desired target style. Formally, $\log P_C(\cdot)$ denotes the class probability predicted by a given style classifier and $\tilde{l}_i$ symbolizes the target style. As such, the translated text can be defined as $\tilde{\mathbf{x}}_i = D(E_s(\mathbf{x}_i), \tilde{l}_i)$. The loss function is formulated in equation 3, where $X \in \{\text{BERT}, \text{CNN}\}$ stands for the classifier used.

$$\begin{aligned} L^{\mathcal{U}}_{\text{style-X}} = - \mathop{\mathbb{E}}_{\mathbf{x}_i \sim \mathcal{U}} \Bigg[& \mathop{\mathbb{E}}_{\hat{\mathbf{x}}_i \sim p_D(\hat{\mathbf{x}}_i \mid E_s(\mathbf{x}_i), l_i)} \log p_C\left(l_i \mid \hat{\mathbf{x}}_i\right) \\ & + \mathop{\mathbb{E}}_{\tilde{\mathbf{x}}_i \sim p_D\left(\tilde{\mathbf{x}}_i \mid E_s(\mathbf{x}_i), \tilde{l}_i\right)} \log p_C\left(\tilde{l}_i \mid \tilde{\mathbf{x}}_i\right) \Bigg] \end{aligned} \tag{3}$$

3.1.3 Back-translation loss. This loss function ensures that the semantics of the original sentence is preserved in the translated text by mapping it back to the original sentence. The translated sentence $\tilde{\mathbf{x}}_i$ is encoded using E_s and is input to the decoder D along with the original style information l_i to reconstruct $\mathbf{x}_i$. As such, the back-translation loss is defined in Equation 4.

$$L^{\mathcal{U}}_{btrans} = - \mathop{\mathbb{E}}_{\mathbf{x}_i \sim \mathcal{U}, \tilde{\mathbf{x}}_i \sim p_D(\tilde{\mathbf{x}}_i \mid E_s(x_i), \tilde{l}_i)} \log p_D\left(\mathbf{x}_i \mid E_s\left(\tilde{\mathbf{x}}_i\right), l_i\right) \tag{4}$$

3.2 Loss sequentiality

Inspired by curriculum learning [2], we propose the idea of loss function sequentiality. Introducing loss functions successively enables the neural network to take into account specific tasks at a time so that they successfully combine for the final task at hand. The underlying idea is that the cognitive process of text style transfer may not be completely parallel, and some sub-tasks such as content preservation and style transfer accuracy may combine in sequence. As a consequence, we propose that each loss function is included in the learning process following a certain periodicity, i.e., after a given number of epochs, $ep \in \{1, 3, 5\}$[2]. As such, the network first learns its weights for a specific loss function (which corresponds to some facet of the task) and builds upon these weights to self-tune for the other loss functions in sequence (each one corresponding to some other facet of the task). Formally, we define a new global loss function $L^{\text{ep}}_{\text{final}}$, which includes individual losses L_{i_q}, such that $i_q \in \{\text{recon}, \text{style}, \text{btrans}\}$, at regular ep epochs as in equation 5.

[2]The periodicity can be any integer, but experimental results show that convergence is quickly attained.

$$L^{\text{ep}}_{\text{final}} = L^{\mathcal{U},0.ep}_{i_0} + L^{\mathcal{U},1.ep}_{i_1} + L^{\mathcal{U},2.ep}_{i_2} \quad (5)$$

3.3 Combination of style classifiers

Most TST architectures implement the pre-trained TextCNN [13] as style classifier. This 1D CNN can be viewed as a local feature extractor emphasizing on n-grams, thus grasping the morpho-syntactic relationships between adjacent words, but eventually failing in accounting for the overall semantics of the input sentence. As a consequence, we propose to incorporate a global feature extractor, which would be able to comprehend the contextualized semantic meaning of the text. For that purpose, we learn a BERT architecture [5] to provide a combined loss function to account for style transfer accuracy, as defined in equation 6.

$$L^{\mathcal{U}}_{\text{style}} = L^{\mathcal{U}}_{\text{style-CNN}} + L^{\mathcal{U}}_{\text{style-BERT}} \quad (6)$$

4 EXPERIMENTAL SETUPS

Dataset. To perform our experiments, we use the Yelp polarity reviews dataset[3]. For training, we consider sentences less than or equal to 15 words in length, and we randomly sample 35k instances such that both positive and negative classes are equally balanced. For a fair comparison with previous works, we report scores on the validation set provided by Li et al. [15].

Implementation details. The sentence encoder and decoder are BERT architectures implemented using the open-source library provided in [26]. Both content and style representations consist of 768 dimensions. They are concatenated and passed through a single feed-forward neural network to retain their original size. Both the BERT and TextCNN style classifiers are pre-trained on the Yelp dataset (weights are frozen during training). Each model is trained for 20 epochs with early stopping criteria. We use AdamW optimizer [16] with a batch size of 8 and a learning rate of 5×10^{-6}. A Gumbel-softmax distribution [10] is used at the decoder, which results in a continuous approximation to discrete sampling, thereby allowing the gradients to back-propagate through the network.

5 EVALUATION RESULTS

Evaluation metrics. We report scores for three automated evaluation metrics, namely style transfer accuracy (STA) [23], self-BLEU score [14] and perplexity (PPL) [14], and two other scores for their combinations, i.e. G2 [17] and GM4, an adaptation of GM [12]. In particular, STA assesses the degree to which accurate style translation is achieved. It is evaluated by TextCNN [13]. The self-BLEU score computes BLEU4 between the original and the translated sentence and accounts for content preservation. Fluency is estimated by the PPL score, which is computed based on the GPT-2 language model [21]. To account for the general behavior of a given model, G2 computes the geometric mean of STA and self-BLEU, while GM4 is the geometric mean of STA, self-BLEU[4], and the inverse of PPL. Two human gold standards have also been proposed by [24] (H) and [17] (H4), that respectively contain one and four human references, respectively, translated by different annotators. As a consequence, we propose the BLEU_H and BLEU_H4 metrics to account for "real" content preservation that corresponds to BLEU4 between the original and target sentences (for BLEU_H4, this is an average BLEU4). Finally, we propose two new metrics, G2_H4 and GM4_H4, that evidence the global behavior of a given system when taking into account "real" content preservation, i.e., instead of relying on self-BLEU as in G2 and GM4, we respectively use BLEU_H and BLEU_H4 instead, to give rise to G2_H4 and GM4_H4.

[3]https://www.tensorflow.org/datasets/catalog/yelp_polarity_reviews

[4]We use BLEU4 to compute self-BLEU, instead of word overlap as in [12] for GM, as it is a more complete text distance metric.

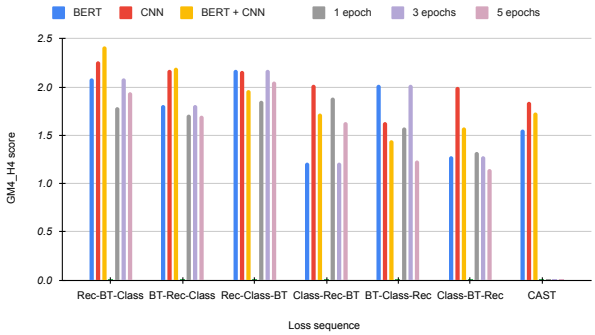

Figure 2: Impact of the style classifier and the number of epochs by loss sequence in terms of GM4_H4 score.

Discussion of the results. Results are illustrated in Table 1 which clearly shows that the introduction of loss sequentiality can significantly boost the performance of a standard TST model and that our best configuration strongly competes with the current SOTA model, i.e., [17]. Indeed, if we take globalized metrics, our CAST-inspired architecture with the loss sequence Rec-BT-Class learned with a periodicity of 3 epochs for a combination of BERT and CNN style classifiers evidences best results in three out of four cases, i.e., G2, GM4, and GM4_H4, while it also shows the second-best result overall in terms of G2_H4. Moreover, if we strictly stand for the comparison with human references (i.e., BLEU_H and BLEU_H4), our best architecture presents third and second-best results, respectively, both for H and H4. Interestingly, the best results over H and H4 are obtained by the TST models proposed by the authors of the respective human annotations. Moreover, [17] do not present results over H. In comparison, we present all the results.

Nevertheless, not all loss sequences perform equally. In particular, introducing the style classification loss in the first epoch is evidently the worst configuration in terms of GM4 and GM4_H4. This result must be mitigated by the fact that higher scores can be obtained if perplexity is not taken into account, i.e., G2 and G2_H4. Indeed, the introduction of the BERT style classifier allows better fluency of the generated sentences when the loss functions are introduced sequentially. Moreover, the architecture without loss sequentiality achieves impressive results for BLEU_H and BLEU_H4, but clearly down performs in terms of perplexity, thus generating odd sentences and evidencing low GM4 and GM4_H4 results.

Impact of periodicity and style classifier. In Figure 2, we show the performance of all the loss sequences in terms of GM4_H4 by periodicity (number of epochs) and style classifier combination (BERT and CNN). Results clearly show the advantage of combining BERT and CNN style classifiers for the Rec-BT-Class and BT-Rec-Class sequences. However, this situation is only valid for these two

Table 1: All results on the Yelp dataset. Figures in bold (resp. underlined) refer to the best (resp. second best) value overall. For our models, only best results are shown for each loss sequence. Rec, BT, Class stand for Reconstruction, Back-Translation and Style Classification losses.

Baseline Models	BLEU_H	BLEU_H4	STA	self-BLEU	PPL	G2	G2_H4	GM4	GM4_H4
Li et al. - Retrieval (2018) [15]	15.00	2.90	97.90	2.60	-	15.95	16.85	-	-
Prabhumoye et al. (2018) [20]	2.00	5.00	95.40	2.80	417	16.34	21.84	0.862	1.046
Fu et al - Style Embedding (2017) [6]	19.20	42.30	8.70	67.40	600	24.22	19.18	0.992	0.850
Vineet et al. - VAE (2019) [12]	-	-	93.00	7.60	**303**	26.59	-	1.326	-
Shen et al. (2017) [23]	7.80	17.90	75.30	20.70	395	39.48	36.71	1.580	1.506
Fu et al - Multidecoder (2017) [6]	12.90	27.90	50.20	40.10	350	44.87	37.42	1.792	1.588
Li et al. - DeleteAndRetrieve (2018) [15]	14.70	32.60	88.90	36.80	318	57.20	53.83	2.175	2.089
Hu et al. (2017) [9]	22.30	-	86.70	58.40	-	71.16	-	-	-
Tian et al. (2018) [24]	**24.90**	-	92.70	63.30	-	76.60	-	-	-
Luo et al. - DualRL (2019) [17]	-	**55.20**	85.60	68.70	457	76.69	68.74	2.343	2.179

Cheng et al. - CAST (2020) [4]		BLEU_H	BLEU_H4	STA	self-BLEU	PPL	G2	G2_H4	GM4	GM4_H4
Loss Sequence	Style classifier									
No sequence	BERT	18.29	34.65	93.00	44.00	846	63.97	56.77	1.691	1.562
No sequence	CNN + BERT	22.22	42.80	99.18	69.24	807	82.87	65.15	2.041	1.739
No sequence	CNN	23.65	45.91	99.15	71.03	723	83.92	67.47	2.136	1.847

Our Models			BLEU_H	BLEU_H4	STA	self-BLEU	PPL	G2	G2_H4	GM4	GM4_H4
Loss sequence	Periodicity	Style classifier									
Class-BT-Rec	1 epoch	CNN	23.55	44.42	98.85	61.28	526	77.83	66.26	2.258	2.028
Class-Rec-BT	3 epochs	CNN	23.60	45.41	**99.63**	64.98	543	80.46	67.26	2.285	2.028
BT-Class-Rec	3 epochs	BERT	18.03	36.07	80.18	54.66	352	66.20	53.78	2.318	2.018
Rec-Class-BT	3 epochs	CNN	24.87	48.48	98.70	74.60	474	85.81	**69.17**	2.495	2.161
BT-Rec-Class	3 epochs	CNN + BERT	22.66	42.62	99.00	66.87	398	81.36	64.96	2.552	2.196
Rec-BT-Class	3 epochs	CNN + BERT	24.81	49.34	96.30	**78.56**	333	**86.98**	68.93	**2.832**	**2.426**

Human baseline (H4)	-	100	74.00	-	-	-	86	-	-

sequences, as, for the other configurations, the CNN and BERT classifiers alone outperform their combination. If the first (resp. second) loss function to be introduced is Class (resp. Class), then the CNN (resp. BERT) classifier is the best performing configuration. If the generated sentence is fluent, the BERT style classifier is likely to correctly account for style accuracy, while the CNN style classifier may produce better results if the generated sentence is odd. This explains that when the first focus in the loss sequence is on content preservation (Rec or BT), results are better with BERT, while if it is on style transfer accuracy (Class), the CNN takes the lead. It is also interesting to note that the combination of BERT and CNN style classifiers improves performance when the Class loss is introduced at the end, thus taking advantage of both style classifiers specifics.

In terms of periodicity, results clearly show the superiority of the configuration, which introduces each loss function after 3 epochs. More experiments have been performed to include a superior number of epochs, but results were steadily decreasing after 5 epochs[5]. Moreover, it is interesting to note that convergence was obtained earlier with loss function sequences compared to gathering all loss functions at the beginning of the learning process.

Some sentence generation examples. Table 2 illustrates typical generation examples for positive-to-negative and negative-to-positive transfers. Qualitative analysis indicates that the adjectives and adverbs are usually correctly handheld for both style transfers, even if two or more positive (resp. negative) items are contained in the same sentence. However, when the text is long, and its semantics is complex due to the concatenation of different sentences, the quality of the transfer decreases as the TST model only takes into account part of the text to be transferred in style. Finally, although most sentences are grammatically correct, there is an overuse of final punctuation, which results in an odd transfer style, and produces inappropriate translated sentences.

Table 2: Generated sentences by our best model.

Original : positive	Generated : negative
Excellent food with great service.	Poor food with terrible service. √
Pretty awesome place. Great pools and kid friendly.	Pretty ok place. Worst pools and kid rude. ⌣
I love this Olive Garden!	I hate this Olive Garden? ⌢
Original : negative	**Generated : positive**
Worst service ever.	Best service ever! √
I love the food. however service here is horrible!	I love the food. however service here is fantastic. ⌣
I hate US Airways, shitty services.	I hate US Airways, shitty services! ⌢

6 CONCLUSION

In this paper, we propose to improve text style transfer by introducing loss functions sequentiality into the learning process, thus following the ideas of curriculum learning. In particular, we show that we can reach SOTA results starting from a baseline model for sentiment transfer over the Yelp dataset. Moreover, we evidence that the combination of distinct style classifiers focusing on different textual information can boost the generation performance. Future work implies the replication of this study over different transfer tasks such as gender [14], formality [7] or offensiveness [19], and the definition of an optimization process to find the best loss sequence. The code is available for reproducibility upon demand[6].

[5]Note that similar behaviors are evidenced for G2, G2_H4, and GM4 both for the impact of periodicity and style classifier combination.

[6]Institution regulation rules do not allow the use of code hosting platforms.

REFERENCES

[1] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural Machine Translation by Jointly Learning to Align and Translate. In *3rd International Conference on Learning Representation (ICLR)*, Yoshua Bengio and Yann LeCun (Eds.).

[2] Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. 2009. Curriculum Learning. In *26th Annual International Conference on Machine Learning (ICML)*. 41–48.

[3] Keith Carlson, Allen Riddell, and Daniel Rockmore. 2018. Evaluating prose style transfer with the Bible. *Royal Society open science* 5, 10 (2018), 171920.

[4] Yu Cheng, Zhe Gan, Yizhe Zhang, Oussama Elachqar, Dianqi Li, and Jingjing Liu. 2020. Contextual Text Style Transfer. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Trevor Cohn, Yulan He, and Yang Liu (Eds.). 2915–2924.

[5] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, Jill Burstein, Christy Doran, and Thamar Solorio (Eds.). 4171–4186.

[6] Zhenxin Fu, Xiaoye Tan, Nanyun Peng, Dongyan Zhao, and Rui Yan. 2018. Style Transfer in Text: Exploration and Evaluation. In *32nd Conference on Artificial Intelligence (AAAI)*. 663–670.

[7] Hongyu Gong, Suma Bhat, Lingfei Wu, JinJun Xiong, and Wen-mei Hwu. 2019. Reinforcement Learning Based Text Style Transfer without Parallel Training Corpus. In *Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*. 3168–3180.

[8] Zhiqiang Hu, Roy Ka-Wei Lee, and Charu C. Aggarwal. 2020. Text Style Transfer: A Review and Experiment Evaluation. *CoRR* abs/2010.12742 (2020). arXiv:2010.12742

[9] Zhiting Hu, Zichao Yang, Xiaodan Liang, Ruslan Salakhutdinov, and Eric P Xing. 2017. Toward controlled generation of text. In *34th International Conference on Machine Learning (ICML)*. 1587–1596.

[10] Eric Jang, Shixiang Gu, and Ben Poole. 2017. Categorical Reparameterization with Gumbel-Softmax. In *5th International Conference on Learning Representations (ICLR)*.

[11] Harsh Jhamtani, Varun Gangal, Eduard Hovy, and Eric Nyberg. 2017. Shakespearizing Modern Language Using Copy-Enriched Sequence to Sequence Models. In *Workshop on Stylistic Variation at 57th Annual Meeting of the Association for Computational Linguistics (ACL)*. 10–19.

[12] Vineet John, Lili Mou, Hareesh Bahuleyan, and Olga Vechtomova. 2019. Disentangled Representation Learning for Non-Parallel Text Style Transfer. In *57th Annual Meeting of the Association for Computational Linguistics (ACL)*. 424–434.

[13] Yoon Kim. 2014. Convolutional Neural Networks for Sentence Classification. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 1746–1751.

[14] Guillaume Lample, Sandeep Subramanian, Eric Michael Smith, Ludovic Denoyer, Marc'Aurelio Ranzato, and Y-Lan Boureau. 2019. Multiple-Attribute Text Rewriting. In *7th International Conference on Learning Representations (ICLR)*.

[15] Juncen Li, Robin Jia, He He, and Percy Liang. 2018. Delete, Retrieve, Generate: a Simple Approach to Sentiment and Style Transfer. In *Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, Marilyn A. Walker, Heng Ji, and Amanda Stent (Eds.). 1865–1874.

[16] Ilya Loshchilov and Frank Hutter. 2019. Decoupled Weight Decay Regularization. In *7th International Conference on Learning Representations (ICLR)*.

[17] Fuli Luo, Peng Li, Jie Zhou, Pengcheng Yang, Baobao Chang, Xu Sun, and Zhifang Sui. 2019. A Dual Reinforcement Learning Framework for Unsupervised Text Style Transfer. In *28th International Joint Conference on Artificial Intelligence (IJCAI)*. 5116–5122.

[18] John Lyons. 1995. *Linguistic Semantics: An Introduction.* Cambridge University Press.

[19] Cicero Nogueira dos Santos, Igor Melnyk, and Inkit Padhi. 2018. Fighting Offensive Language on Social Media with Unsupervised Text Style Transfer. In *56th Annual Meeting of the Association for Computational Linguistics (ACL)*. 189–194.

[20] Shrimai Prabhumoye, Yulia Tsvetkov, Ruslan Salakhutdinov, and Alan W Black. 2018. Style Transfer Through Back-Translation. In *56th Annual Meeting of the Association for Computational Linguistics (ACL)*. 866–876.

[21] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language Models are Unsupervised Multitask Learners. (2019). https://openai.com/blog/better-language-models/

[22] Parker Riley, Noah Constant, Mandy Guo, G. Kumar, David C. Uthus, and Zarana Parekh. 2020. TextSETTR: Label-Free Text Style Extraction and Tunable Targeted Restyling. *ArXiv* abs/2010.03802 (2020).

[23] Tianxiao Shen, Tao Lei, Regina Barzilay, and Tommi Jaakkola. 2017. Style Transfer from Non-Parallel Text by Cross-Alignment. In *Advances in Neural Information Processing Systems (NeurIPS)*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (Eds.).

[24] Youzhi Tian, Zhiting Hu, and Zhou Yu. 2018. Structured Content Preservation for Unsupervised Text Style Transfer. *CoRR* abs/1810.06526 (2018). arXiv:1810.06526

[25] Tsung-Hsien Wen, David Vandyke, Nikola Mrkšić, Milica Gašić, Lina M. Rojas-Barahona, Pei-Hao Su, Stefan Ultes, and Steve Young. 2017. A Network-based End-to-End Trainable Task-oriented Dialogue System. In *15th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*. 438–449.

[26] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. HuggingFace's Transformers: State-of-the-Art Natural Language Processing. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 38–45.

[27] Chen Wu, Xuancheng Ren, Fuli Luo, and Xu Sun. 2019. A Hierarchical Reinforced Sequence Operation Method for Unsupervised Text Style Transfer. In *57th Conference of the Association for Computational Linguistics (ACL)*, Anna Korhonen, David R. Traum, and Lluís Màrquez (Eds.). 4873–4883.

[28] Xing Wu, Tao Zhang, Liangjun Zang, Jizhong Han, and Songlin Hu. 2019. Mask and Infill: Applying Masked Language Model for Sentiment Transfer. In *28hth International Joint Conference on Artificial Intelligence (IJCAI)*, Sarit Kraus (Ed.). 5271–5277.

[29] Jingjing Xu, Xu Sun, Qi Zeng, Xiaodong Zhang, Xuancheng Ren, Houfeng Wang, and Wenjie Li. 2018. Unpaired Sentiment-to-Sentiment Translation: A Cycled Reinforcement Learning Approach. In *56th Annual Meeting of the Association for Computational Linguistics (ACL)*, Iryna Gurevych and Yusuke Miyao (Eds.). 979–988.

[30] Peng Xu, Jackie Chi Kit Cheung, and Yanshuai Cao. 2020. On variational learning of controllable representations for text without supervision. In *International Conference on Machine Learning (ICML)*. 10534–10543.

[31] Zichao Yang, Zhiting Hu, Chris Dyer, Eric P Xing, and Taylor Berg-Kirkpatrick. 2018. Unsupervised Text Style Transfer using Language Models as Discriminators. In *Advances in Neural Information Processing Systems (NeurIPS)*.

[32] Yi Zhang, Jingjing Xu, Pengcheng Yang, and Xu Sun. 2018. Learning Sentiment Memories for Sentiment Modification without Parallel Data. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii (Eds.). 1103–1108.

[33] Junbo Zhao, Yoon Kim, Kelly Zhang, Alexander Rush, and Yann LeCun. 2018. Adversarially Regularized Autoencoders. In *37th International Conference on Machine Learning (ICML)*. 5902–5911.