Event-Centered Prompting for Text Style Transfer

徐勝 [1] 鈴木良弥 [2] 福本文代 [2]
山梨大学大学院 医工農学総合教育部 [1] 総合研究部工学域 [2]
{g22dts03,ysuzuki,fukumoto}@yamanashi.ac.jp

Abstract

Text style transfer (TST), recently, attracted significant research interest. Although previous attempts have shown outstanding performance for sentiment or formality transfer, they still encounter limitations to some extent considering the arbitrariness of context and the lack of annotated corpora. In this work, we explore an **E**vent-**C**entered **P**rompt (**ECP**) strategy leading to causal large language model (LLM) transferring the text style. Based on the strategy, we explore the inference of LLM by two steps of the prompt: the model retrieves the event of the source text along the prompt, then further generates the target following another prompt. Our ECP strategy can be applicable to design prompt formats for diverse style transfer tasks. The experimental results on the Yelp and Amazon datasets which leverage the LLaMA as the backbone show the effectiveness of our method. The source code is available online[1)].

1 Introduction

With the rapid growth of deep learning techniques, significant progress has been made in the field of language models and language generation research fields in recent years. However, text style transfer (TST) and controllable generation are still challenging to infer plausible text due to human-centered demands. The main objective of TST is to transfer the style of a given text while reserving the style-isolated content. From the early explores [1] [2] to recent works [3] [4], TST is still one of the interesting topics.

There are three main paradigms of resolutions for TST tasks. One attempt is to explore the generation models extensively. Those include the variational auto-encoder (VAE) [5] [6] [7], generative adversarial network (GAN) [8] [9], and diffusion model [4]. The second paradigm is fine-tuning approaches that show the outstanding capability of diverse transformer derivations. For instance, Lai et.al augmented the BART [10] and GPT2 [11] with two rewards strategies (content and style) [12]. Hu et.al designed a framework by connecting neural adapters capturing different style information for multi-attribute TST [13]. The third paradigm is prompt-based methods which have also attracted much attention, especially the large language model (LLM) such as the GPT-4[14] and Gemini[15] obtained a breakthrough in open-ended text generation. Reif et.al proposed augmented zero-shot learning to lead LLM transfer arbitrary style [16]. Suzgun et.al further improved the quality of inference by a re-ranking strategy [17].

However, the attempts to leverage generation models are still suffering from limitations to some extent. Compared with the strong generalization capacity of PLM on huge non-parallel corpora the generation techniques exhibit shortcomings in complex context and different domain. The fine-tuning strategies also suffer from the lack of expensive annotated datasets. Moreover, the prompt-based works [16] [17] regarded TST as the seq2seq task and straightforwardly explore the relationship between the source and target. They didn't explicitly consider the connection between the center event shared by the source and the target text.

In this paper, we define the center event and utilize it as the connection between a pair sentence with opposite sentiment styles and explore an event-centered prompt strategy to explicitly lead LLM to transfer from source to target step-by-step. Following Allan et al work, we define the event as something that happens at some specific time and place along with all necessary preconditions and unavoidable consequences [18]. In the TST task, considering that the expected target shares the same style-isolated content with the source, we assume that they are two paraphrases with different styles from one central event describing the core information and contexts.

1) https://github.com/codesedoc/ecp

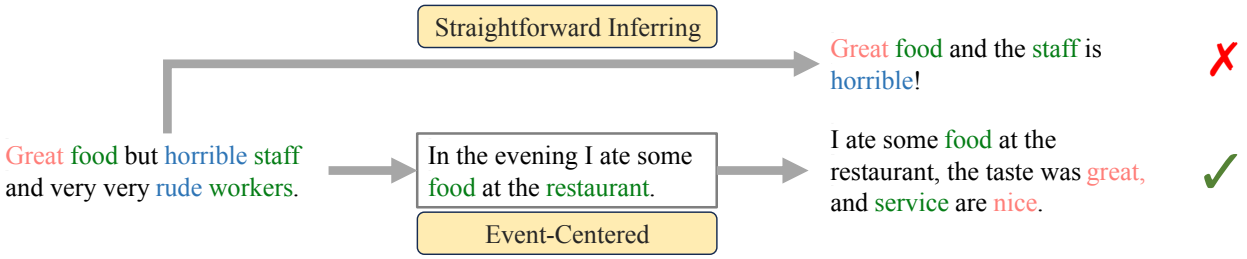

Figure 1 Our ECP strategy comparing with the straightforward seq2seq paradigm. In the example sentence, the content marked with green color represents the core information in the source, target, and center event. The blue and pink font indicates the text with negative and positive sentiments, respectively.

2 Event-Centered Prompting

Figure 1 illustrates our CEP strategy by using a simple example from the negative to the positive style transfer. The input source is selected from the Yelp dataset, and the straightforward transfer is inferred by the language model. As we can see the straightforward transfer suffers from the hallucination issue. This obstacle could be caused by the conflict of the opposite "Great" and "horrible" from the source sentence.

To overcome this issue, we leverage the event by combining the core information and context as the connection between source and target to assist the model in inferring expected generations. One proper description of the center event can be explained as "In the evening I ate some food at the restaurant" which shows only information about the fact. Based on this description/context without style, we assume that the model can generate a more accurate target style.

2.1 Problem Formulation

Let $S = \{s_1, s_2, ..., s_t\}$ represent the style of t available values interested in sepcific TST task. In this paper, we focus on the sentiment style transfer, the S can be regarded as a binary set $\{neg, pos\}$, where the neg and pos are positive and negative styles, respectively. We consider two main transfer cases, i.e., from *positive* to *negative* and from *negative* to *positive* ($pos \rightleftarrows neg$). Given a pair of source text X, and its target counterpart Y with a expire style label $s \in S$, e.g., *positive*, the objective of the TST task is formulated as the language model $\mathbb{P}(\mathrm{Y}|\mathrm{X}, s)$. We can view the model as an optimization problem on the specific dataset by utilizing extensive neural networks as the backbone.

We assume that one proper description of the event, marked as E, exists and it combines the necessary context information and core content shared by X and Y. The TST task can be further decomposed as follows:

$$
\begin{aligned}
\mathbb{P}(\mathrm{Y}|\mathrm{X}, s) &= \frac{\mathbb{P}(\mathrm{Y}, \mathrm{X}, s)}{\mathbb{P}(\mathrm{X}, s)} \\
&\geqslant \frac{\mathbb{P}(\mathrm{Y}, \mathrm{X}, \mathrm{E}, s)}{\mathbb{P}(\mathrm{X}, s)} \\
&= \frac{\mathbb{P}(\mathrm{Y}, \mathrm{X}, \mathrm{E}, s)}{\mathbb{P}(\mathrm{X})\,\mathbb{P}(s)} = \frac{\mathbb{P}(\mathrm{X}, \mathrm{E})}{\mathbb{P}(\mathrm{X})} \cdot \frac{\mathbb{P}(\mathrm{Y}, \mathrm{X}, \mathrm{E}, s)}{\mathbb{P}(\mathrm{X}, \mathrm{E})\,\mathbb{P}(s)} \\
&= \frac{\mathbb{P}(\mathrm{X}, \mathrm{E})}{\mathbb{P}(\mathrm{X})} \cdot \frac{\mathbb{P}(\mathrm{Y}, \mathrm{X}, \mathrm{E}, s)}{\mathbb{P}(\mathrm{X}, \mathrm{E}, s)} \\
&= \underbrace{\mathbb{P}(\mathrm{E}|\mathrm{X})}_{\textit{reducation}}\ \underbrace{\mathbb{P}(\mathrm{Y}|\mathrm{X}, \mathrm{E}, s)}_{\textit{synthesis}} \qquad (1)
\end{aligned}
$$

Following the language model shown in Eq. (1), the optimization of the objective of the TST task can be decomposed into two components with lower bounds. We call each reduction and synthesis, respectively. Theoretically, those available candidates for resolving the original optimization problem as the previous attempts can also be extended to the two new sub-problems.

2.2 Reduction and Synthesis

Note that the autoregressive pre-train objective is more inherently similar to the optimization components of Eq. (1) and outstanding performance for open-end text generation. We thus prompt the LLM to infer a proper description of the event from the source. We call this procedure *reduction* step. We then further lead the model to generate the expected target by another prompt, called *synthesis* step. Inspired by [19], the reduction and synthesis can be regarded as guidance that helps the pre-trained language model to transfer the sentiment polarity of the source sequence step-by-step.

Let P_r and P_s represent the prompt formats for *reduction*

Data	Transfer	Model	Acc	r-sBLEU	s-sBLEU	t-PPL	s-PPL
Yelp	$neg \rightarrow pos$	LLaMA-0s	52.6	14.5	31.7	34	123
		LLaMA-1s	**66.2**	10.3	20.4	**24**	56
		ECP-0s	42.0	**20.2**	**49.7**	47	143
		ECP-1s	58.0	12.2	26.6	25	**55**
	$pos \rightarrow neg$	LLaMA-0s	70.8	15.5	37.7	49	173
		LLaMA-1s	76.4	13.4	33.7	35	86
		ECP-0s	69.0	**19.2**	**49.4**	62	191
		ECP-1s	**79.4**	13.1	34.8	**33**	**77**
Amazon	$neg \rightarrow pos$	LLaMA-0s	51.6	21.0	38.7	40	112
		LLaMA-1s	**61.6**	14.1	25.7	**28**	**66**
		ECP-0s	41.0	**29.8**	**56.0**	60	172
		ECP-1s	58.4	17.7	31.2	31	**66**
	$pos \rightarrow neg$	LLaMA-0s	53.4	29.0	48.2	53	139
		LLaMA-1s	63.4	25.4	42.0	**41**	93
		ECP-0s	51.6	**35.3**	**60.0**	72	193
		ECP-1s	**66.2**	26.1	46.0	**41**	**67**

Table 1 Comparison with the baselines on Yelp and Amazon datasets. ECP-0s and ECP-1s indicate zero-shot, and one-shot-based inference, respectively. LLaMA-0s and LLaMA-1s are the baselines. Bold font refers to the best result in each style transfer.

and *synthesis*, respectively, and $\mathcal{F}_{LLM}$ indicates the inference process of LLM. The description of event E from source X can be obtained by Eq. (2).

$$\mathrm{E} = \mathcal{F}_{LLM}(P_r(\mathrm{X})) \tag{2}$$

The final generation Y is inferred as Eq. (3).

$$\mathrm{Y} = \mathcal{F}_{LLM}(P_s(\mathrm{X}, \mathrm{E}, s)) \tag{3}$$

We name the two-step strategy for the TST task shown as Eq. (2) and Eq. (3) as event-centered prompting.

3 Experiments

3.1 Experimental settings

We chose the LLaMA-7B [20] as the backbone LLM for the time and memory cost, which is running on a single 48GB GPU: NVIDIA RTX A6000. The max generation length is set to 1,024. Following the experiments reported by [17], we chose curly brackets as the text's delimiter, and the format of the prompt is set as "Contrastive". The temperature of the last softmax layer is set to 0.8. The token decoder strategy is nucleus sampling with a threshold of 0.95.

3.2 Data and evaluation metrics

We conducted experiments on the two popular datasets for SST, Yelp reviews [21] and Amazon reviews [22]. For comparing with related works, we utilize the version of these two datasets cleaned by [17]. Table 2 lists the size of the test set, i.e., the total number of sentence pairs, and the average length of the token sequence.

Much of the previous work evaluated their methods by using three evaluation metrics, content preservation, transfer strength, and fluency. For a fair comparison with the baselines, we also used these metrics. The first is content preservation which consists of reference-BLEU (r-sBLEU) and self-BLEU (s-sBLEU). The sBLEU means the SacreBLEU scores following the setting of [17]. Here, r-sBLEU measures the distance of generated sentences from the ground-truth references, and s-sBLEU refers to the degree to which the model directly copies the source.

The second is transfer strength, which is scored by using accuracy on the target style of the generations. The last is the fluency of generated texts. We calculated the average token-level perplexity (t-PPL) and average sentence-level perplexity (s-PPL) for generated texts. For calculating the SacreBLEU and PPL scores, we leverage the evalu-

Dataset	Size	Avg. Len.
Yelp	1,000	13.9
Amazon	1,000	16.7

Table 2 The statistics of Yelp and Amazon dataset

ator which is available from the Hugging Face[2]. The gpt2-large is selected as the backbone to compute the PPL scores. A python toolkit for sentiment analysis, named pysentimiento[3] [23] is utilized to load a text classifier to obtain the accuracy.

3.3 Results

Table 1 shows comparative results against the baseline obtained by LLaMA [17] on Yelp and Amazon datasets. Overall, our model is better than the baseline on the Yelp dataset in both transfer types ($pos \rightleftarrows neg$) and all evaluation metrics except for LLaMA-1s by Acc and t-PPL. The improvement compared with the baseline, LLaMa is 0~3.9%, 0~39.3%, 3.2~ 44.8%, 0~ 38.2%, and 0~10.4% for Acc, r-sBLEU, s-sBLEU, t-PPL, and s-PPL, respectively. Likewise, our model on Amazon is better than the baseline except for LLaMA-1s by Acc and t-PPL. [17] reported that $neg \rightarrow pos$ sentiment style transfer is more challenging for LLMs. The improvement compared with the baseline, LLaMa is 0~4.2%, 1.8~29.5%, 8.7~ 30.9%, 0~ 33.3%, and 0~38.8% for Acc, r-sBLEU, s-sBLEU, t-PPL, and s-PPL, respectively.

We also observed that the results obtained by $pos \rightarrow neg$ transfer are consistently better than $neg \rightarrow pos$ transfer in both datasets. The differences between zero-shot and one-shot performance obtained by our approach are larger than those obtained by the baseline. This indicates that our prompt strategy that leverages only one example can assist the LLM to lead consistent performance gain across all evaluation metrics.

Table 3 shows the comparative results against related works on Yelp data, and the sentiment transfer is $pos \rightarrow neg$. We can see that similar to Suzgun et al's models, LLaMA, we compare on par or favorably despite using much smaller models. Specifically, our model with the zero-shot is better than other zero-shot models in all of the evaluation metrics except for LLaMA-0s by Acc, especially, the improvement compared with the second best, LLaMA-0s is 23.9%, 31.0%, and 26.5% for r-sBLEU, s-sBLEU, and t-PPL, respectively. Table 3 also indicates that our model with the one-shot is slightly worse than LLaMa-1s on r-sBLEU and t-PPL while there are no significant differences between them.

Model	Acc†	r-sBLEU	s-sBLEU	t-PPL
Related Works				
[1] LLM-0s	-	5.3	9.2	33
[1] LLM-5s	-	6.7	11.2	43
[2] GPT-J-6B-0s	-	14.3	34.7	49
[2] GPT-J-6B-4s	-	25.3	50.5	107
Baselines				
[3] LLaMA-0s	70.8	15.5	37.7	49
[3] LLaMA-1s	76.4	13.4	33.7	35
ECP				
ECP-0s	69.0	19.2	49.4	62
ECP-1s	79.4	13.1	34.8	33

Table 3 A comparison with related works on the Yelp dataset. ECP-0s and ECP-1s refer to zero-shot, and one-shot-based inference, respectively. LLaMA-0s and LLaMA-1s are the baselines that infer the generation with the same contrastive prompt of [3] Touvron et al. [20]. References: [1] Reif et al. approach [16], [2] Suzgun et al. [17], [3] Touvron et al. [20]. Note on †: we used pysentimiento to obtain the Acc score which is different from [17].

4 Conclusion

We proposed a few-shot learning strategy that generates a target style via two-step prompts: reduction to mine style-free sequence from the input text, and synthesis to change the target style to generate the output text. Experimental results on Yelp and Amazon review datasets showed that our model is comparable to the baseline, LLaMA with both zero and one-shot-based inference, especially it works well on content preservation, r-sBLEU and s-sBLEU metrics. There are interesting directions for future work. To examine the performance against the scales of LLMs, we will apply our model to LLaMA-13B and 30B. To evaluate the robustness of our proposed method, we also apply our model to other style transfers such as politics and topic mentioned in [24]. For further improvement, we are going to extend our ECP by prompt learnable method to control the text generation inspired from the works of [25].

2) https://huggingface.co/docs/evaluate/index
3) https://github.com/pysentimiento/pysentimiento

Acknowledgement

This work was supported by JST, the establishment of university fellowships towards the creation of science technology innovation, Grant Number JPMJFS2117, SCAT, and JKA.

References

[1] David D. McDonald and James D. Pustejovsky. A computational theory of prose style for natural language generation. In **Second Conference of the EACL**, pp. 187–193, 1985.

[2] Eduard Hovy. Generating natural language under pragmatic constraints. **Journal of Pragmatics**, Vol. 11, No. 6, pp. 689–719, 1987.

[3] Jingxuan Han, Quan Wang, Licheng Zhang, Weidong Chen, Yan Song, and Zhendong Mao. Text style transfer with contrastive transfer pattern mining. In **Proc. of the 61st Annual Meeting of the ACL (Volume 1: Long Papers)**, pp. 7914–7927, 2023.

[4] Yiwei Lyu, Tiange Luo, Jiacheng Shi, Todd Hollon, and Honglak Lee. Fine-grained text style transfer with diffusion-based language models. In **Proc. of the 8th Workshop on Representation Learning for NLP (RepL4NLP 2023)**, pp. 65–74, 2023.

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

[6] Vineet John, Lili Mou, Hareesh Bahuleyan, and Olga Vechtomova. Disentangled representation learning for non-parallel text style transfer. In **Proc. of the 57th Annual Meeting of the ACL**, pp. 424–434, 2019.

[7] Yu Bao, Hao Zhou, Shujian Huang, Lei Li, Lili Mou, Olga Vechtomova, Xin-yu Dai, and Jiajun Chen. Generating sentences from disentangled syntactic and semantic spaces. In **Proc. of the 57th Annual Meeting of the ACL**, pp. 6008–6019, 2019.

[8] Tianxiao Shen, Tao Lei, Regina Barzilay, and Tommi Jaakkola. Style transfer from non-parallel text by cross-alignment. In **Advances in Neural Information Processing Systems**, Vol. 30, 2017.

[9] Junbo Zhao, Yoon Kim, Kelly Zhang, Alexander Rush, and Yann LeCun. Adversarially regularized autoencoders. In **Proc. of the 35th International Conference on Machine Learning**, pp. 5902–5911, 2018.

[10] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In **Proc. of the 58th Annual Meeting of the ACL**, pp. 7871–7880, 2020.

[11] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. In **OpenAI blog 1(8):9**, 2019.

[12] Huiyuan Lai, Antonio Toral, and Malvina Nissim. Thank you BART! rewarding pre-trained models improves formality style transfer. In **Proc. of the 59th Annual Meeting of the ACL and the 11th IJCNLP (Volume 2: Short Papers)**, pp. 484–494, 2021.

[13] Zhiqiang Hu, Nancy Chen, and Roy Lee. Adapter-TST: A parameter efficient method for multiple-attribute text style transfer. In **Findings of the Association for Computational Linguistics: EMNLP 2023**, pp. 693–703, 2023.

[14] OpenAI. Gpt-4 technical report, 2023.

[15] Google Gemini Team. Gemini: A family of highly capable multimodal models, 2023.

[16] Emily Reif, Daphne Ippolito, Ann Yuan, Andy Coenen, Chris Callison-Burch, and Jason Wei. A recipe for arbitrary text style transfer with large language models. In **Proc. of the 60th Annual Meeting of the ACL (Volume 2: Short Papers)**, pp. 837–848, 2022.

[17] Mirac Suzgun, Luke Melas-Kyriazi, and Dan Jurafsky. Prompt-and-rerank: A method for zero-shot and few-shot arbitrary textual style transfer with small language models. In **Proc. of the 2022 Conference on Empirical Methods in Natural Language Processing**, pp. 2195–2222, 2022.

[18] James Allan, editor. **Topic Detection and Tracking, Event-based Information Organization**. Kluwer Academic Publisher, 2002.

[19] Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are zero-shot reasoners. **arXiv preprint arXiv:2205.11916**, 2022.

[20] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. Llama: Open and efficient foundation language models, 2023.

[21] Zhang Xiang, Zhao Junbo, and LeCun Yann. Character-level convolutional networks for text classification. **Advances in Neural Information Processing Systems**, Vol. 28, pp. 649–657, 2015.

[22] Julian McAuley, Christopher Targett, Qinfeng Shi, and Anton van den Hengel. Image-based recommendations on styles and substitutes, 2015.

[23] Juan Manuel Pérez, Juan Carlos Giudici, and Franco Luque. pysentimiento: A python toolkit for sentiment analysis and socialnlp tasks, 2021.

[24] Di Jin, Zhijing Jin, Zhiting Hu, Olga Vechtomova, and Rada Mihalcea. Deep learning for text style transfer: A survey. **Computational Linguistics**, Vol. 48, No. 1, pp. 155–205, 2022.

[25] Kexin Yang, Dayiheng Liu, Wenqiang Lei, Baosong Yang, Mingfeng Xue, Boxing Chen, and Jun Xie. Tailor: A soft-prompt-based approach to attribute-based controlled text generation. In **Proc. of the 61st Annual Meeting of the ACL (Volume 1: Long Papers)**, pp. 410–427, 2023.