Text Style Transfer with Parameter-efficient LLM Finetuning and Round-trip Translation

Anonymous ACL submission

Abstract

This paper proposes a novel method for Text Style Transfer (TST) based on parameter-efficient finetuning of Large Language Models (LLMs). Addressing the scarcity of parallel corpora that map between styles, the study employs round-trip translation to synthesize such parallel datasets from monolingual corpora. This approach creates "neutralized" text devoid of stylistic attributes, essentially creating a shared input style at training-time and inference-time. Experimental results demonstrate consistent superiority of this method over zero-shot prompting and few-shot ICL techniques measured by BLEU scores and style accuracy scores across 6 investigated domains. Furthermore, the integration of retrieval-augmented generation (RAG) for terminology and name knowledge enhances robustness and stylistic consistency.

1 Introduction

Text Style Transfer (TST) is the task of rephrasing text by modifying stylistic attributes while preserving its core attribute-independent semantics and intent (Shen et al., 2017; Toshevska and Gievska, 2024). These stylistic attributes encompass formality, attitude, verbosity, preferred terminology, and other characteristics inherent to the text. A significant challenge in TST lies in the scarcity of annotated parallel corpora, which hinders the application of fully supervised learning or finetuning methods (Pan et al., 2024) in most text style domains.

Roundtrip translation is a machine translation technique where a sentence is translated from one language to a pivot language and then back to the original language. It has been previously used to evaluate MT system robustness and generation quality (Somers, 2005; Moon et al., 2020). Prior work on TST has observed that round-trip translating a sentence effectively diminishes stylistic attributes specific to the author, yielding a "neutralized and generative" style while retaining the content (Sennrich et al., 2016; Rabinovich et al., 2017). This observation motivates the use of roundtrip-translation pipelines as autoencoders in many encoder-decoder styled TST frameworks to extract destylized latent vectors from input text, so that style-specific decoders can be trained in a supervised fashion even when input style domains are unpredictable (Prabhumoye et al., 2018).

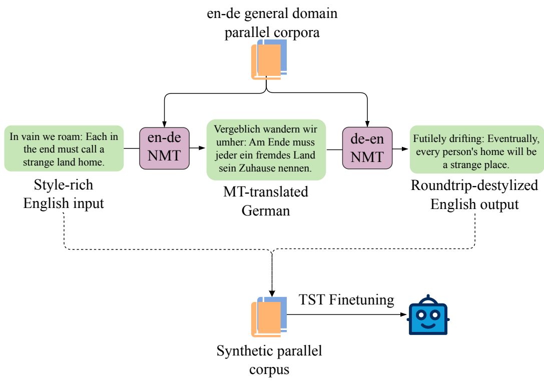

Figure 1: Our proposed workflow for finetuning large language models (LLMs) for text style transfer (TST) using only non-parallel dataset in the target domain. A bilingual general-domain parallel dataset is used to train a pair of neural machine translation (NMT) models capable of translating between English and a pivot language. We then obtain machine-translated style-neutral texts of the original in-domain texts by roundtrip translating the in-domain set with the NMT models. This enables supervised finetuning of LLMs for TST, where we finetune LLMs for MT-output-domain to target-domain transfer using the synthetic parallel corpus.

In this paper, we propose a novel TST method that adapts LLMs for style transfer tasks using non-parallel in-domain corpora and roundtrip translation (Figure 1). Our workflow involves first training two neural machine translation models that serve as the roundtrip translation pipeline, using large-scaled general-domain bilingual parallel cor-

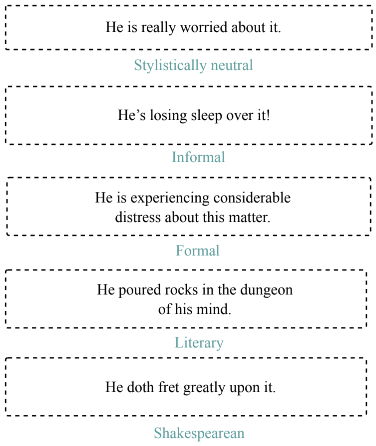

Figure 2: Example sentences illustrating semantically equivalent content in various styles. Outputs of our roundtrip translation pipeline is considered as stylistically neutral.

pora. We then roundtrip translate a monolingual, stylistically consistent corpus using the pretrained NMT models to construct a style-neutral to target-domain pseudo-parallel corpus. This corpus can thus be used to finetune LLMs for TST tasks. Furthermore, to enhance the model's robustness to unseen or complex style domains, we implemented an inference-time workflow that roundtrip translates queries before doing inference, improving training and inference time coherence (§3.1).

We experimented (§4) our style transfer method on several text style domains with distinctive style features (§4.1.1), and compared its performance against two state-of-the-art methods: Few-shot In Context Learning (ICL) and Automatic Post-Editing (APE) (Liu et al., 2024b; Moon et al., 2022). Following prior research, style transfer quality is evaluated using BERT-based style classifiers trained on held-out data and the BLEU score (Subramanian et al., 2018; Wan et al., 2023; Aycock and Bawden, 2024). Our **main contributions** are:

- **Pseudo-parallel dataset construction** (§3.1). We propose a roundtrip translation method for generation synthetic parallel corpus, enabling TST with supervised finetuning in domains lacking bitext.

- **Methods for TST-finetuning** (§4). We systematically evaluate finetuned TST-LLMs employing several different models, prompts, RAG methods, and inference methods, compared against state-of-the-art baselines.

- **Retireval augmentation for finetuning and inference** (§3.2.3). We propose the use of retrieval-augmentation for **finetuning**, carefully experimented with RAG in both finetuning and inference prompts, and validate its effectiveness beyond prompting.

2 Related Work

Supervised TST Several parallel corpora for TST have been released (Voigt et al., 2018; Rao and Tetreault, 2018) that motivated supervised TST on these pre-selected domains with sufficient parallel data, such as Jhamtani et al. (2017)'s work on Shakespearizing modern English. This approach is limited to domains with parallel corpora.

Unsupervised / Semi-supervised Text Style Transfer Due to the scarcity of parallel TST data in most domains, one major focus of prior TST research (Lai et al., 2021; Hu et al., 2022; Nouri, 2022) is the seq2seq encoder-decoder models for unsupervised training with non-parallel target-side data. Central to these frameworks are effective disentanglement of latent representations of styles (Nangi et al., 2021; Voigt et al., 2018) and the preservation of original content through the TST pipeline (Tian et al., 2018). There is recent work on UTST frameworks using LLM prompting and attention masking (Pan et al., 2024).

Roundtrip translation for TST Prior works observed that roundtrip translation tend to reduce authors' stylistic features while preserving the style-independent content (Prabhumoye et al., 2018; Rabinovich et al., 2017). This observation motivates the use of roundtrip-translation as auto-encoders to extract destylized latent vectors from text with various input style domains that represent content. Style-specific decodes then transform these latent vectors to output texts with the same content and the target style (Prabhumoye et al., 2018; Riley et al., 2021). In these settings, roundtrip translation is believed to transform instances in various domains to the same latent representation, essentially turning the task of transferring from varying domains to the simpler task of decoding destylized latent vectors to target style generation, which can be achieved in a supervised fashion.

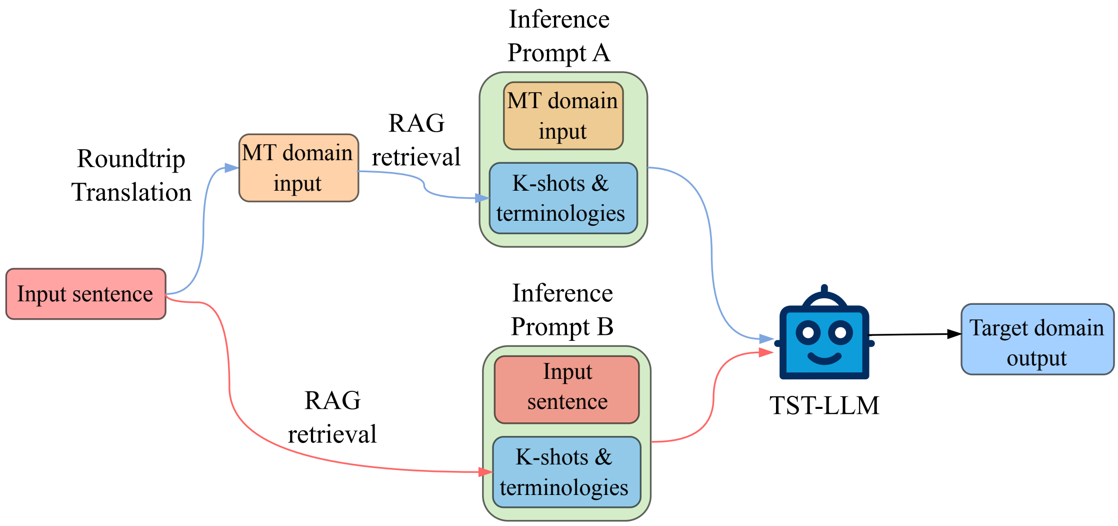

Figure 3: **Our proposed workflow.** We show **two inference routes** that we tested on: **route i** (blue in figure) involves first roundtrip translate the input to match the training-time input domains and then perform RAG-enhanced TST-LLM inference with two retrievers we built (§3.2) on the intermediary text, where as **route ii** (red in figure) directly performs RAG-enhanced TST-LLM inference using the original input. Controlled experiments on these methods demonstrate that roundtrip translating the input first significantly enhances model's performance, bringing especially considerable improvements facing stylistically diverse and complex queries. Findings in this experiment are described in §4.4.

.

LLM-supported TST Recent research indicates that state-of-the-art Large Language Models (LLMs) possess the capability to perform TST tasks when appropriately prompted or finetuned (Liu et al., 2024a; Zhang et al., 2024; Mukherjee et al., 2024). Prior works have developed prompt learning methods for TST that use non-parallel data(Liu et al., 2024a; Wan et al., 2023; Aycock and Bawden, 2024; Zhang et al., 2024). These strategies typically involve augmenting prompts with retrieved data (Liu et al., 2024b; Zhang et al., 2024) and a limited set of in-domain, non-parallel examples ("shots") (Chen, 2024; Liu et al., 2024a; Bhandarkar et al., 2024) in optimized prompt configurations (Liu et al., 2024a). However, these methods are limited to prompts, lacking the ability to introduce parameter-level adjustments that could enhance LLM adaptability to specific TST or domain adaptation contexts. **Parameter-efficient finetuning for TST** has been investigated very recently (Liu et al., 2024b; Mukherjee et al., 2024), but only limited to domains with existing parallel corpora.

3 Methods

3.1 Roundtrip Translation

We propose a novel TST framework that adapts LLMs for style transfer tasks using only nonparallel in-domain corpora (Figure 3). We first train a pair of neural machine translation (NMT) models using Marian (Junczys-Dowmunt et al., 2018) and a large-scale generic bilingual corpus between English and a selected pivot language. This pair of generic NMT models constitutes a roundtrip translation pipeline, which reduces stylistic features of input texts with rich and diverse styles to roundtrip translated style-neutral output. A monolingual style-consistent dataset is roundtrip translated to form a pseudo-parallel dataset, and then we finetune a LLM on this dataset to specialize in MT-destylized to in-domain style transfer tasks.

A potential issue with our method is that, during finetuning we essentially provide **RT-destylized domain** to target domain supervision, rather than **arbitrary domain** to target domain supervision. We make such distinction since machine-translated texts tend to be neutralized and style homogenized, whereas arbitrary inference-time inputs may not exhibit such feature. To mitigate this issue, we designed an inference-time workflow where the inference sentence is also roundtrip-translated to its stylistically neutral counterpart before queried to the finetuned LLM. We compared direct inference and our RT-first inference method in the experiments section (§4.4), and report that RT-first inference yields noticeably better generation quality when dealing with unseen text domains.

3.2 RAG Retrievers for TST-LLM

Lewis et al. (2020) proposed a RAG framework in which a retriever-generator model is trained end-to-end to enhance coherence between the pre-trained retriever and generator subsystems. Inspired by this approach, we incorporate RAG into both the fine-tuning and inference stages of our TST-LLM approach to enhance the LLM's adaption to retrieval-enhanced prompts at inference-time, unlike previous TST-with-LLM methods where RAG is primarily considered a prompting technique (Liu et al., 2024a; Wan et al., 2023; Aycock and Bawden, 2024; Zhang et al., 2024) and finetuning experiments are largely limited to the zero-shot strategy with various prompt templates (Liu et al., 2024b; Mukherjee et al., 2024). In §4.3, we present a comparative implementation of retrieval augmentation at both training time and inference time, demonstrating that RAG also yields significant improvements when applied during training.

3.2.1 Training-time similarity-based example retrieval

Our example retrieval augmentation method involves obtaining sentence pairs as instructions for how we would like the query to be transferred. In order for the example transfer sentence pairs to be instructive, we adapt a similarity-based retrieval method (Figure 4) to retrieve transfer sentence pairs that are similar to the task objective, using cosine distance obtained through the Faiss vectorization liberary (Douze et al., 2024).

Since we provide **sentence pairs** as examples, an issue that naturally arises is whether to provide example pairs whose **source-sides** are similar to the query or those **target-sides** are similar to the expected output. We consider this distinction necessary and vital to the quality of the retrieved shots. Consider the informal phrase "I'm good". Transferring it to the formal domain would have many valid answers, such as "I do not require anything further", "I am content with the arrangement", or straightforwardly "I appreciate your concern; I am in good health." Searching with target-side would at least provide **correct** answers, if not helpful, whereas searching with source-side would potentially yield misleading examples.

Essentially this is the difference between searching with the **questions** and searching with the **answers**. At training-time, providing examples pairs with relevant target-sides is rather straightforward, since the actual output text (or the "completion") is present. We constructed a Faiss vector bank for the monolingual in-domain corpus. Then, for each instance in the pseudo-parallel dataset obtained from roundtrip translation, we take its target-side text, search for top-k most similar examples excluding itself, and look up the source-side counterparts of these retrieved sentences to form example transfer sentence pairs to be put into finetuning prompts.

3.2.2 Inference-time "sketch-first" example retrieval

At inference time when only the out-of-domain-side input is present, we follow Wang et al. (2022)'s schema to use a "sketch-first" example retrieval augmentation logic (Figure 4). We first perform few-shot inference with **randomly-selected** examples to generate a sketch output that resembles the in-domain transferred generation, though with limited quality due to the randomly selected shots. We then use the sketch as the query to retrieve examples with high similarity from the Faiss vector bank to enhance the second-round inference that yields the refined output. In §4.4, we report on inference-time experiments on the inference-time example retrieval augmentation methods described above and the RT-first inference pipeline described in (§3.1).

3.2.3 Terminology and Name List Retrieval

Diction and word preferences are an important aspect of text style domains. The same concept or object can be referred to by different terminologies in different domains, such as football in British English and soccer in American English, so consistently using the correct terminology for the target domain is vital for semantic correctness and style consistency. In literary-translation domains, there is a similar issue of **naming consistency**, where machine-translated works may use semantic translations and direct translations in different contexts to refer to the same characters, causing confusion and inconsistency.

We improve our TST model's terminology correctness and long-term consistency by retrieving

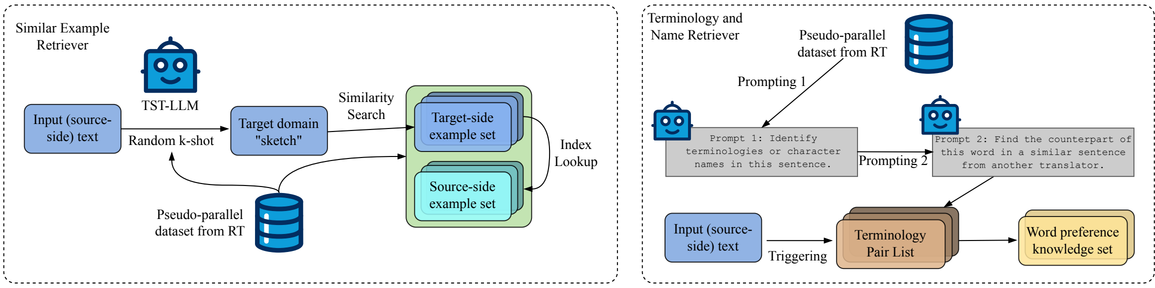

Figure 4: **Retrieval augmentation workflow. Left (a):** Similarity-based example retriever. We vectorize and index **the target-side texts** of the parallel synthetic datasets for nearest-neighbor search. For each query, we first do k-shot inference with the finetuned TST-LLM to obtain an "in-domain" sketch, which is used as search query in the target-side dataset to obtain k most similar pairs. Note that this is for **inference-time RAG**. For finetuning prompts, we can search with the target side texts directly without the need for an in-domain sketch. **Right (b)** Terminology and name retriever: For each instance in the synthetic parallel datasets, the first LLM call extracts relevant words from the source side, then the second call matches them with their counterparts in the target side, yielding a terminology pair list for each domain. During inference, each input is checked against these term pairs; where relevant matches are found, a concise guiding sentence is appended to the prompt.

a **terminology and name list** from our pseudo-parallel corpus, and add relevant terminology instructions to prompts when some trigger words are present in the query (Figure 4). For each object in the pseudo-parallel corpus, we first prompt a LLM with the source side of the paired sentences and ask it to identify any terminologies or names in it. Then, we do a second round of prompting with the target side of the sentence pairs alongside the retrieved terminologies, and ask the LLM to find their counterparts. Through this pipeline, we construct a list of source domain to target domain preferred terminologies pairs. If any of the source-side words are present in the query, we add a one-sentence instruction in the inference prompt that provides terminology and name transfer guidance. Prompts we used are in Appendix A.

3.3 Parameter Efficient LoRA Finetuning

LoRA (Low-Rank Adaptation of Large Language Models) is an efficient approach that reduces computational and memory costs by using low-rank approximation techniques (Hu et al., 2021). The LoRA approach involves freezing the pre-trained model's weight matrices and introducing trainable low-rank decomposition matrices into the model's layers. This approach allows us to finetune the 7B and 8B LLMs with 2 NVIDIA A100 GPUs, each with 81GB of memory. Hyperparameters and configurations we used are put in Appendix B.

3.4 Evaluation: BLEU and style classification accuracy score

We primarily evaluate two aspects of our models, namely style transfer quality and content preservation ability. We train a BERT-based (Devlin et al., 2018) style classifier for each style domain using held-out in-domain data, in the same fashion as Liu et al. (2024b,a); Mukherjee et al. (2024)'s prior works. The trained classifier classifies a given text to be either in-domain or out-of-domain, thus the generation from our TST models is tested with these classifiers to yield a style classification accuracy, as a measure of how well the generated texts aligns with the target domain in terms of text style. BLEU scores between generation and source texts are used to evaluate to what extent the original meaning is preserved after transfer.

4 Experiments

4.1 Experiment Setup

4.1.1 Datasets and Synthetic Data Generation

A large-scale generic parallel training set is used to train the Neural Machine Translation model pairs for each pivot language. We used Marian(Junczys-Dowmunt et al., 2018) for these Neural Machine Translation models. Detailed configurations we used are given in Appendix B. Several monolingual style-consistent corpora are roundtrip translated to construct pseudo-parallel datasets for finetuning. Our experiments encompassed six distinct stylistic domains: corpus of administrative documentation from the Internal Revenue Service (IRS), corpus of

Dataset	Language	# Sentence	# Word
WMT24	en–de	75,991,652	1,160,839,966
WMT24	en–zh	72,192,512	857,631,464
IRS	nonparallel en	455,733	7,349,231
Treasury	nonparallel en	408,004	8,990,216
NCBI	nonparallel en	201,888	3,509,166
Dailymed	nonparallel en	496,013	9,154,195
Literary	nonparallel en	105,030	3,643,974
Webnovel	nonparallel en	1,000,000	16,498,331

Table 1: **Datasets.** The WMT24 datasets are used to train generic NMT models for roundtrip-translation. We selected Chinese and German as the pivot languages. These datasets are obtained through WMT24 general MT track training set releases. The IRS dataset, the Treasury dataset, the Dailymed dataset, and the NCBI dataset are crawled from their respective websites. The literary dataset is a collection of English translations of pre-modern Asian literary works. The webnovel dataset is released by the WMT24 literary MT track.

official communication corpus from the U.S. Department of Treasury, medical literature from the Dailymed database, scientific publications from the National Center for Biotechnology Information (NCBI) Database, the corpus of literary translations of pre-modern Chinese texts by six productive translator, including David Hawkes and John Minford, and a corpus of human-translated Chinese web novels from WMT 2024. These domain-specific corpora served as the foundation for creating parallel finetuning datasets.

4.1.2 TST Prompt Templates

We experimented on three potential prompt templates for TST finetuning. Prompt details and experiments on prompts are in Appendix A. After careful examination we decided to use the prompt template in Figure 5 throughout our experiments.

Rewrite the given sentence into the style of [style name].
Here are [n] examples:
Input: [example input i]. Output: [example output i].
......
Note that you may want to rewrite "[input terminology]" to "[output terminology]" for contextual consistency.
Now go ahead: Input: [query input]. The [style name] output:

Figure 5: The prompt template we use for Text Style Transfer Finetuning. Performances of other prompts that we experimented on are put in Appendix A.

4.2 Experiments on Pretrained LLMs

We experimented on various LLMs to evaluate their potentials for TST finetuning with synthetic parallel data. For all models, we performed sketch-first 5-shot finetuning without any other knowledge retrieval. A BERT classifier is trained for each text style domain and used on the generated text to yield the style accuracy score for each experimental group. Results are shown in Table 2.

Out of the four models we investigated, **Llama-3-8B-Instruct** and **Gorilla-openfunctions-v2** have the best overall performances across the four tested style domains, with the finetuned Gorilla LLM having the highest average BLEU score and the finetuned Llama-3 LLM having the highest average style accuracy score. We will use Llama-3-8B-Instruct as the base model for prompting and finetuning for other experiments in the rest of this section.

4.3 Experiments on Retrieval Augmentation Methods

Here we present the experiment results with regards to various RAG methods that we used during both finetuning and inference (Table 3). The random k-shot example retrieval method retrieves k random **pairs** of style-neutral to target-domain sentences for each finetuning prompt and each inference prompt (Figure 5). Similar k-shot method retrieve the k most similar examples pairs, which is achieved through **direct cosine distance search** at finetuning time, and through **sketch-first method** (§3.2.2) at inference time. Terminology and name retrieval are achieved by constructing a **terminology pair bank** (§3.2.3).

Note that these groups in Table 3 are using different finetuning methods *and* different inference methods, since we also include the retrieved information in the finetuning prompts. Sketch-first similar 5-shot finetuning consistently outperforms the prompting and zero-shot finetuning baselines across the four tested domains, with a highest BLEU score of 52.35 and highest Style Accuracy score of 0.865 both in the Pre-modern Literary domain. The affect of example retrieval on the BLEU score is more consistent and stable that its affect on the style classification accuracy. For style classification accuracy, the similar 5-shot model is still predominantly the best-performing model, though random 3-shot and 5-shot models have a 0.030 - 0.037 higher classification acc. in the IRS domain

Pretrained LLMs	IRS domain		Literary domain		Treasury domain		NCBI domain	
	BLEU	Acc.	BLEU	Acc.	BLEU	Acc.	BLEU	Acc.
Baseline	22.53	0.391	21.90	0.172	24.15	0.245	19.87	0.354
meta-llama/Llama-3.1-8B-Instruct	**48.89**	**0.826**	41.42	**0.721**	45.22	**0.812**	46.30	**0.896**
gorilla-llm/gorilla-openfunctions-v2	47.40	0.756	**42.31**	0.663	**47.80**	0.714	**49.62**	0.823
mistralai/Mistral-7B iii	43.30	0.742	36.85	0.701	40.12	0.710	38.43	0.734
facebook/opt-2.7b	38.12	0.640	35.15	0.570	42.00	0.820	41.27	0.676

Table 2: TST Finetuning performance with Various Base LLMs (random 5-shot instructions finetuning). All four tested models exhibit strong potential in performing TST tasks with proper finetuning, with Llama-3.1-8B-Instruct and Gorrila-openfunctions-v2 having considerablly higher performance in both content preservation and style adaptation across the four tested domains.

RAG methods	IRS domain		Literary domain		Treasury domain		NCBI domain	
	BLEU	Acc.	BLEU	Acc.	BLEU	Acc.	BLEU	Acc.
5-shot ICL	27.79	0.591	25.90	0.613	24.72	0.541	27.87	0.462
APE with Marian	36.81	0.642	35.72	0.649	36.37	0.621	35.95	0.659
Zero-shot finetuning	42.39	0.793	40.39	0.742	41.43	0.826	39.30	0.742
Random 3-shot finetuning	47.23	0.839	39.96	0.732	44.41	0.796	42.07	0.823
Random 5-shot finetuning	48.89	0.826	41.42	0.721	45.22	0.812	46.30	**0.896**
Similar 3-shot finetuning	47.79	0.749	48.83	0.812	47.79	0.820	49.01	0.776
Similar 5-shot finetuning	**49.50**	0.796	52.35	0.865	**50.46**	0.876	49.96	0.831
5-shot ICL w/ terminology and name retrieval	28.53	0.672	26.25	0.669	26.69	0.729	29.31	0.586
Similar 5-shot finetuning w/ terminology and name retrieval	49.28	**0.895**	**52.61**	**0.933**	50.25	**0.894**	**50.37**	0.872

Table 3: TST performance with various retrieval augmentation methods and scale (Using Llama-3.1-8B-Instruct). The ICL method prompts the LLM with k in-domain example sentences as context knowledge. Random k-shot finetuning provides random examples at both finetuning and inference time; Similar k-shot provides similar examples for finetuning prompts through cosine distance search, and for inference prompts in a sketch-first manner (§3.2.2). Terminology and name retrieval constructs a term pair bank, which is added to the prompt when triggered (§3.2.3). Providing LLMs with examples at both training and inference time brings considerable improvements, especially when providing **similar** examples. 5-shot groups tend to have stronger effects on both BLEU score and Acc. than 3-shot and 0-shot.

and the NCBI health domain. We attribute this to the fact that the IRS and NCBI domains **are closer to the general domain** than the Literary and Treasury domains, making the classification of generated texts for these domains more nuanced and unpredictable.

Looking into the generated text across the experimental groups and the style domains, we observed that similarity-based n-shot finetuning is much more stable than random n-shot finetuning, especially for the Literary domain, where sentence length, diction, and phrasing habits vary to a great extent throughout the corpus. When provided with irrelevant examples at inference time, such as one word long sentence examples for long discourses or descriptive sentences provided as examples for character speeches, the examples can even mislead the model and lower the generation quality compared to zero-shot inference. Similarity-based 3-shot and 5-shot finetuning, on the other hand, exhibits a much more stable improvement in generation quality, as it always provides examples with similar length and overlapping words with the query sentence. It yields up to 12.22 increase in BLEU score and 0.191 increase in style classification accuracy across the four tested style domains.

We also observed that terminology and name retrieval has stronger influence on prompting than on the finetuning – adding the terminology paraphrase guidance results in a 7.29% average improvement on the Acc. score for 5-shot finetuning, and a 18.62% average improvement on the Acc.

Inference methods	IRS domain		Literary domain		Treasury domain		NCBI domain	
	BLEU	Acc.	BLEU	Acc.	BLEU	Acc.	BLEU	Acc.
0-shot inference	43.21	0.811	46.68	0.842	42.25	0.742	46.63	0.696
RT & random 5-shot inference	45.53	0.809	47.12	0.792	43.31	0.782	45.51	0.742
similar 5-shot inference	**48.73**	0.829	52.33	0.820	**50.47**	0.833	49.96	0.793
RT & similar 5-shot inference	46.28	**0.895**	**51.61**	**0.933**	50.25	**0.894**	**50.37**	**0.872**

Table 4: TST Finetuning performance with various inference-time workflows. All groups are inferences with a LLama3.1-8B-instruct that is finetuned with similar 5-shot and terminology RA from the previous experiment(§4.3). 0-shot inference uses prompts that do not provide any additional knowledge besides task description. RT-first inferences means we roundtrip translate the queries to match finetuning input domains(§3.1) before being given to the LLMs. Results suggest a significant boost in style classification accuracy brought by RT-first and similar shots, and a moderate improvement in BLEU score brought by similar shots.

score for 5-shot ICL.

4.4 Experiments on Inference methods

We also conducted controlled experiments on various inference-time workflows. All inference groups utilized the LLama3.1-8B-instruct model, finetuned with the same 5-shot approach. **They only differ in inference methods.** The 0-shot inference setting employed inference prompts containing only task descriptions without additional knowledge. The RT-first inference method involved roundtrip translation (RT) of queries to align with the finetuning input domain (§3.1) before feeding them into the LLM. The similar k-shot inference method retrieves and provides relevant examples in a sketch-first manner, as elaborated in §3.2.2.

Results indicate that RT-first and similar-shot approaches both bring significant enhancements to style classification accuracy, while similar-shot inference also yields a moderate improvement in BLEU score. However, we observed that roundtrip translation can reduce BLEU scores, suggesting potential semantic drift when queries are mapped to the MT-output style neutral domain. The extent of this information loss is likely influenced by several factors, including **pivot language selection**, the **quality of NMT models**, and the **complexity of the style domain**. Despite this trade-off, the substantial improvement in style classification accuracy underscores the importance of the RT-first workflow.

5 Conclusion

This study has established a robust method for Text Style Transfer (TST) that leverages parameter-efficient finetuning of Large Language Models (LLMs) combined with round-trip translation to address the challenges posed by the scarcity of parallel corpora in most stylistic domains. Through round-trip translation, we produce synthesized pseudo-parallel texts that reconstruct a supervised Text Style Transfer setting from MT-neutralized domain to target style domain. The MT-neutralized style domain serves as a shared input style, so that inputs with unseen stylistic features better match the finetuned LLM at inference time, enhancing adaptability and robustness when facing out-of-domain input sentences. Our experiments across six distinct style domains demonstrate that the round-trip translation augmented finetuning method consistently outperforms state-of-the-art approaches, such as In-Context Learning and Automatic Post-Editing for TST.

We also found that retrieval-augmented generation (RAG) effectively enhances terminology and name consistency within our roundtrip translation augmented finetuning framework. Our comprehensive experiments show that incorporating retrieved examples and generation guidance helps maintain long-term stylistic consistency and improves overall generation quality. These findings demonstrate that the application of knowledge and example retrieval augmentation can go beyond prompting.

Our TST finetuning method has the potential to extend beyond single-domain adaptation. Future work could explore multi-style transfer within a single finetuned LLM and investigate more nuanced, non-binary style transfer tasks, such as formality editing.

Limitations

The main limitations of our work are as follows:

- **Semantic drift and error propagation.** Our method relies on machine translation models to generate parallel finetuning datasets. As a result, its performance depends on the quality of the underlying NMT systems and their training data. We observed that when these models introduce errors or cause semantic drift during roundtrip translation, such inaccuracies become embedded in the synthetic parallel corpus used for finetuning. We applied post-processing steps to mitigate such effects, and further efforts could also be made to test various NMT methods or architectures to find the most ideal configuration for the TST task. These improvements and post-editing works, however, are beyond the scope of this study.

- **Alternatives for the Current Roundtrip Translation Pipeline**. In this work, we primarily used Marian to train the NMT models and did not explore alternative methods or workflows for performing roundtrip translation. An intriguing potential alternative is to employ large language models to perform machine translation, either by ICL or finetuning, which might yield better results compared to the current Marian-based approach. However, we did not test these alternative approaches in the current study due to the limit of time and length.

- **Limits to domains with available corpus**. Due to data availability constraints, our experiments are conducted on six style domains, which may not fully capture the range of stylistic variations encountered in real-world scenarios. This limitation could introduce biases into our analysis and potentially restrict the generalizability of our methods. We selected domains that are as diverse and distinctive as possible—from literary to governmental and medical texts—in an effort to enhance the overall robustness and applicability of our method. We strive to enhance the generalizability of our experiments and demonstrate the effectiveness of our method in different domains and conditions.

References

Seth Aycock and Rachel Bawden. 2024. Topic-guided example selection for domain adaptation in llm-based machine translation.

Avanti Bhandarkar, Ronald Wilson, Anushka Swarup, and Damon Woodard. 2024. Emulating author style: A feasibility study of prompt-enabled text stylization with off-the-shelf llms.

Jianlin Chen. 2024. Lmstyle benchmark: Evaluating text style transfer for chatbots.

Yue Chen, Chen Huang, Yang Deng, Wenqiang Lei, Dingnan Jin, Jia Liu, and Tat-Seng Chua. 2024. Style: Improving domain transferability of asking clarification questions in large language model powered conversational agents.

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

Matthijs Douze, Alexandr Guzhva, Chengqi Deng, Jeff Johnson, Gergely Szilvasy, Pierre-Emmanuel Mazaré, Maria Lomeli, Lucas Hosseini, and Hervé Jégou. 2024. The faiss library. *Preprint*, arxiv:2401.08281 [cs].

Johannes Eschbach-Dymanus, Frank Essenberger, Bianka Buschbeck, and Miriam Exel. 2024. Exploring the effectiveness of llm domain adaptation for business it machine translation.

Dimitar F do Carmo, D Shterionov, Joss Moorkens, Joachim Wagner, Murhaf Hossari, Eric Paquin, Dag Schmidtke, Declan Groves, and Andy Way. 2021. A review of the state-of-the-art in automatic post-editing. *Machine Translation*, 35:101–143.

Junxian He, Xinyi Wang, Graham Neubig, and Taylor Berg-Kirkpatrick. 2020. A probabilistic formulation of unsupervised text style transfer.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

Zhiqiang Hu, Roy Ka-Wei Lee, Charu C. Aggarwal, and Aston Zhang. 2022. Text style transfer: A review and experimental evaluation. *SIGKDD Explor. Newsl.*, 24(1):14–45.

Harsh Jhamtani, Varun Gangal, Eduard Hovy, and Eric Nyberg. 2017. Shakespearizing modern language using copy-enriched sequence to sequence models. In *Proceedings of the Workshop on Stylistic Variation*, pages 10–19, Copenhagen, Denmark. Association for Computational Linguistics.

Marcin Junczys-Dowmunt, Roman Grundkiewicz, Tomasz Dwojak, Hieu Hoang, Kenneth Heafield, Tom Neckermann, Frank Seide, Ulrich Germann, Alham Fikri Aji, Nikolay Bogoychev, André F. T. Martins, and Alexandra Birch. 2018. Marian: Fast neural machine translation in c++.

Huiyuan Lai, Antonio Toral, and Malvina Nissim. 2021. Thank you BART! rewarding pre-trained models improves formality style transfer. In *Proceedings of the*

59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 484–494, Online. Association for Computational Linguistics.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems*, volume 33, pages 9459–9474. Curran Associates, Inc.

Jinpeng Li, Zekai Zhang, Quan Tu, Xin Cheng, Dongyan Zhao, and Rui Yan. 2024. Stylechat: Learning recitation-augmented memory in llms for stylized dialogue generation.

Qingyi Liu, Jinghui Qin, Wenxuan Ye, Hao Mou, Yuxuan He, and Keze Wang. 2024a. Adaptive prompt routing for arbitrary text style transfer with pre-trained language models.

Xinyue Liu, Harshita Diddee, and Daphne Ippolito. 2024b. Customizing large language model generation style using parameter-efficient finetuning.

Yinhong Liu, Yimai Fang, David Vandyke, and Nigel Collier. 2024c. Toad: Task-oriented automatic dialogs with diverse response styles.

Hyeonseok Moon, Chanjun Park, Jaehyung Seo, Sugyeong Eo, and Heuiseok Lim. 2022. An automatic post editing with efficient and simple data generation method. *IEEE Access*, 10:21032–21040.

Jihyung Moon, Hyunchang Cho, and Eunjeong L. Park. 2020. Revisiting round-trip translation for quality estimation. *CoRR*, abs/2004.13937.

Sourabrata Mukherjee, Atul Kr Ojha, and Ondřej Dušek. 2024. Are large language models actually good at text style transfer?

Sharmila Reddy Nangi, Niyati Chhaya, Sopan Khosla, Nikhil Kaushik, and Harshit Nyati. 2021. Counterfactuals to control latent disentangled text representations for style transfer.

Jianmo Ni, Jiacheng Li, and Julian Mcauley. 2019. Justifying recommendations using distantly-labeled reviews and fine-grained aspects.

Nasim Nouri. 2022. Text style transfer via optimal transport.

WMT24 Organizers. 2024. Findings of the 2024 conference on machine translation (WMT24). In *Proceedings of the 2024 Conference on Machine Translation*, TBD. Association for Computational Linguistics.

Lei Pan, Yunshi Lan, Yang Li, and Weining Qian. 2024. Unsupervised text style transfer via llms and attention masking with multi-way interactions.

Shishir G. Patil, Tianjun Zhang, Xin Wang, and Joseph E. Gonzalez. 2023. Gorilla: Large language model connected with massive apis. *Preprint*, arXiv:2305.15334.

Shrimai Prabhumoye, Yulia Tsvetkov, Ruslan Salakhutdinov, and Alan W Black. 2018. Style transfer through back-translation.

Ella Rabinovich, Raj Nath Patel, Shachar Mirkin, Lucia Specia, and Shuly Wintner. 2017. Personalized machine translation: Preserving original author traits. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 1074–1084, Valencia, Spain. Association for Computational Linguistics.

Sudha Rao and Joel Tetreault. 2018. Dear sir or madam, may I introduce the GYAFC dataset: Corpus, benchmarks and metrics for formality style transfer. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 129–140, New Orleans, Louisiana. Association for Computational Linguistics.

Emily Reif, Daphne Ippolito, Ann Yuan, Andy Coenen, Chris Callison-Burch, and Jason Wei. 2021. A recipe for arbitrary text style transfer with large language models.

Parker Riley, Noah Constant, Mandy Guo, Girish Kumar, David Uthus, and Zarana Parekh. 2021. Textsettr: Few-shot text style extraction and tunable targeted restyling.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Improving neural machine translation models with monolingual data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 86–96, Berlin, Germany. Association for Computational Linguistics.

Tianxiao Shen, Tao Lei, Regina Barzilay, Tommi Jaakkola, and Mit Csail. 2017. Style transfer from non-parallel text by cross-alignment.

Harold Somers. 2005. Round-trip translation: What is it good for? In *Proceedings of the Australasian Language Technology Workshop 2005*, pages 127–133.

Sandeep Subramanian, Guillaume Lample, Eric Michael Smith, Ludovic Denoyer, Marc'Aurelio Ranzato, and Y-Lan Boureau. 2018. Multiple-attribute text style transfer.

Zhen Tao, Dinghao Xi, Zhiyu Li, Liumin Tang, and Wei Xu. 2024. Cat-llm: Prompting large language models with text style definition for chinese article-style transfer.

Youzhi Tian, Zhiting Hu, and Zhou Yu. 2018. Structured content preservation for unsupervised text style transfer.

Antonio Toral and Andy Way. 2018. *What Level of Quality Can Neural Machine Translation Attain on Literary Text?*, pages 263–287. Springer International Publishing, Cham.

Martina Toshevska and Sonja Gievska. 2024. *Large Language Models for Text Style Transfer: Exploratory Analysis of Prompting and Knowledge Augmentation Techniques*.

Rob Voigt, David Jurgens, Vinodkumar Prabhakaran, Dan Jurafsky, and Yulia Tsvetkov. 2018. Rtgender: A corpus for studying differential responses to gender.

Zhen Wan, Yating Zhang, Yexiang Wang, Fei Cheng, and Sadao Kurohashi. 2023. Reformulating domain adaptation of large language models as adapt-retrieve-revise: A case study on chinese legal domain.

Yifan Wang, Zewei Sun, Shanbo Cheng, Weiguo Zheng, and Mingxuan Wang. 2022. Controlling styles in neural machine translation with activation prompt.

Chiyu Zhang, Honglong Cai, Yuezhang, Li, Yuexin Wu, Le Hou, and Muhammad Abdul-Mageed. 2024. Distilling text style transfer with self-explanation from llms.

Zhirui Zhang, Shuo Ren, Shujie Liu, Jianyong Wang, Peng Chen, Mu Li, Ming Zhou, and Enhong Chen. 2018. Style transfer as unsupervised machine translation.

A Prompt Templates

TST finetuning prompts:

We experimented on three potential prompt templates for text style transfer (TST) finetuning with synthetic parallel data (Table 5). These prompts organize the query input sentence and several example sentence pairs into a prompt, with proper task descriptions and guidance for the generation. Template (I) and (II) explicitly states the rewriting task, but have different orders of the example and query content. Template (III) is a classic Machine Translation prompt template with demonstrated effectiveness for Machine Translation with LLM. By changing language name to style domain name, we adapt it to guide LLM for text style transfer task.

In this experiment (Table 6), we conducted random 5-shot finetuning with terminology retrieval on Llama3.1-8B-Instruct with the different templates, while leaving other conditions unchanged. Template (I) has the overall highest score in the two tested domains. This is potentially because the query input in template (I) is closer to the end, while in the second template there are many examples separating the query input and the expected generation output. The phrasing of the text style transfer task in prompt (I) is also more ideal than the simplified version in template (III) and better describes the task. Noticeably, template (III), though simple and concise, also has consistently high style accuracy scores in the tested domains.

Table 5: Prompts for TST finetuning

Prompt Template Index	Prompt Template Text
I	Rewrite the following sentence into the style of [style name]. Here are [n] examples: Input: [example input i]. Output: [example output i]. Note that word [input terminology] should be rewritten to [output terminology] for contextual consistency. Now go ahead: Input: [query input]. The [style name] output:
II	Rewrite the following sentence into the style of [style name]: Input: [query input]. Here are [n] examples: Input: [example input i]. Output: [example output i]. Note that word [input terminology] should be rewritten to [output terminology] for contextual consistency.
III	Note that word [input terminology] should be rewritten to [output terminology] for contextual consistency. General domain: [example input i]. [style name] domain: [example output i]. general domain: Input: [query input]. [style name] domain:

Candidate prompt templates for LLM style transfer finetuning. [example input i] and [example output i] indicate the ith pair of retrieved similar examples. In zero-shot finetuning and inferences these lines are removed from the template. [input terminology i] and [export terminology i] are the ith pair of terminologies on the terminology list that is relevant to this inference. [style name] indicate the one-word name for the text style we are adapting to. The naming of style domains are less signifcant for finetuning.

Terminology RAG prompts:

We retrieved terminology and name pair lists for each domain to enhance TST performances, by calling the LLM twice for each instance in the synthetic parallel corpus. The prompts we used are shown in Table 7.

B Hyperparameters and experiment configurations

LoRA finetuning hyperparameters:

We set the learning rate to 2e-4, rank for the low-rank approximation is set to 512, the scaling

Template	IRS domain		Literary domain	
	BLEU	Acc.	BLEU	Acc.
Baseline	22.53	0.391	21.90	0.172
Template I	**48.89**	**0.826**	**41.42**	0.721
Template II	45.40	0.542	38.29	0.563
Template III	46.28	0.781	37.71	**0.794**

Table 6: BLEU and acc. score across IRS and Literary domains for three potential templates. Template (I) has consistently higher BLEU score compared to template (II) and (III), indicating superior ability in content-preservation. Both Template (I) and (III) have stablly high style classification accuracy, indicating robust ability in transferring to target style. In general, the effect of phrasing in prompt templates on TST performance is relatively mild, with template (I) being the most ideal template amid the tested three.

Table 7: Prompts for terminology RAG

Prompt type	Prompt Text
First round	Identify terminologies or character names in the sentence and return in comma separated format, without any additional explanation. Sentence: [source-side sentence]. Terminologies and names:
Second round	Find the counterpart of the word [source-side retrieved word] in the following sentence and return a single word, without any additional explanation. Sentence: [target-side sentence]:

Prompts for terminology RAG. The first prompt retrieves a list of terminologies and names from the source side sentence of each parallel instance, and for each of these retrieved words, the second prompt retrieves its counterpart in the corresponding target side sentence.

factor is set to 256, and we use float16 data type. A dropout rate of 0.05 is applied. We save and evaluate the model every 2000 steps.

Marian Configurations:

We used Marian for the Neural Machine Translation model pairs in the round-trip translation system. In our system we used the Transformer architecture with R2L Reranking, with learning rate 0.0001, 49500 BPE operations, and step size 20000.