Evaluating Prompt-based Text Style Transfer

Huangrui Chu
Yale Graduate School of Art and Science
huangrui.chu@yale.edu

David Peng
Yale University
david.peng@yale.edu

CPSC 670 Final Project Report, Spring 2023

Abstract

Text style transfer is the task of transferring a text from one written style to another while preserving content. Previous researchers have made use of encoder-decoder architectures to paraphrase a given text into a normalized sentence, then performed stylization on a fine-tuned inverse-paraphraser to insert a specific style. While promising, the method is limited by having to fine-tune separate models for each desired style. Recent work has explored prompt-based text style transfer with pre-trained large language models, allowing more complex transformations to arbitrary styles. We propose that prompt-based text style transfer could be good enough to allow for arbitrary styles while providing reliable prompts. By experimenting with straight prompting as well as two-step paraphrase prompting methods, we benchmark large language models on standard text style transfer methods.[1]

1 Introduction

Humans write in different styles for varying contexts. A text message from a friend might read more informally than a professional email to a colleague, and famous authors are known for their unique writing "voices". Textual styles can be characterized by attributes like formality, politeness, sentiment, emotion, etc. (Jin et al., 2022). At a lower level, features that create differences in style include vocabulary and token frequencies, while more abstract differences include choices in grammar, syntax, and the overall structure of the text. By writing in a specific style, we communicate ideas in personalized ways.

Text style transfer is the task of transferring a text from one written style to another while preserving content, as in Figure 1. In recent work in the adjacent fields of image and audio style transfer, users input a photo and get a rerendering in the style of Van Gogh[2], or users input a voice recording and get an audio clip that sounds like Drake[3]. The future hope is to find methods for a similar process in text style transfer, where users might input a written piece and get a rewritten version in the writing "voice" of Mark Twain.

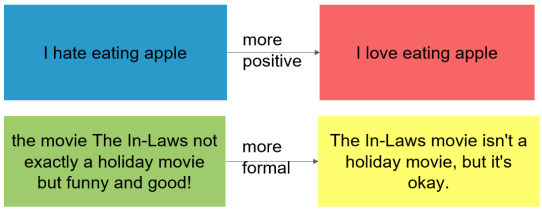

Figure 1: An example of text style transfer for sentiment and for formality.

Two definitions of "style" versus "content" derive from linguistics and practical data: using linguistics, style consists of non-functional linguistic features and content is the semantic meaning. While using data, style is the variance between two corpora and content is invariance (Mou and Vechtomova, 2020).

Unlike in image and audio generation, however, text generation is discrete, which makes back-propagation of output error challenging (Iqbal and Qureshi, 2022). Text style transfer has thus mostly tested transferring style between sentences rather than paragraphs or longer forms of text.

Whereas previous methods like Hu et al. (2017) encode into and decode from a shared latent space, large language models are capable of prompt-based learning and generalizing to many natural language processing tasks, including text style transfer (Brown et al., 2020). Recent work has experimented with prompt-based text style transfer on ver-

[1] Our code and results are public at https://github.com/HuangruiChu/Eval-Prompt-based-TST

[2] https://replicate.com/collections/style-transfer

[3] https://github.com/svc-develop-team/so-vits-svc

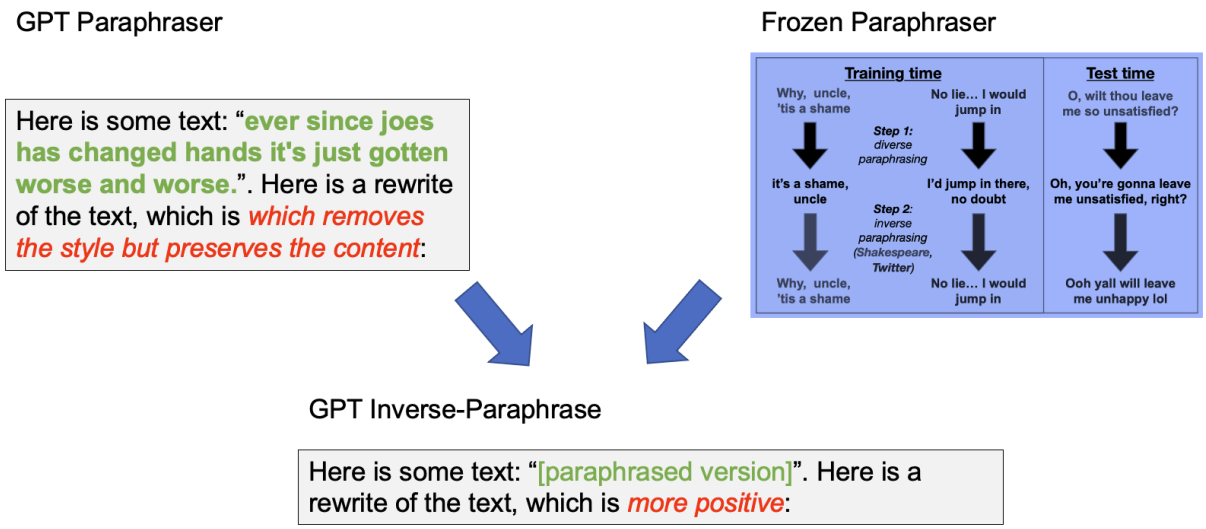

Figure 2: Our proposed method for two-step prompting combines the two-step idea from Krishna et al. (2020) and the prompt-based text style transfer from Reif et al. (2022). First, a paraphraser removes style while preserving context. Then, the paraphrased text is passed through the inverse-paraphraser, which adds the target style to the text. To paraphrase, we can use either a prompt-based method on GPT-3.5 or a frozen paraphraser from Krishna et al. (2020). To allow for arbitrary style transfer, we use GPT-3.5 as the inverse paraphraser.

sions of the latest available large language model, specifically GPT-3, but as far as we are aware, no work has yet benchmarked the newer, fine-tuned version, GPT-3.5 (Ouyang et al., 2022) on standard text style transfer methods.

In this paper, we benchmark the prompt-based performance of GPT-3.5. We also contribute a prompt-based two-step method of style transfer using paraphrasing as a middle layer.

2 Related Work

Krishna et al. (2020) propose a two-step style transfer method using a paraphraser and inverse-paraphraser GPT-2 models. First, the authors train a paraphrasing model that removes the style from a text and keeps only the content, thus normalizing the sentence. Their training set for the normalizer model is curated to maximize lexical and syntactic diversity. Then, for each dataset in their corpora with different desired styles, they train an inverse paraphraser. The idea is that the output of an input text run through the paraphrase and its inverse paraphrase will match the input text. All models are fine-tuned on their respective sole task of removing style or adding style back into a sentence. The overview of their idea is shown in Figure 3.

Reif et al. (2022) pioneer prompt-based style transfer using large language models. They propose three methods: zero-shot; few-shot, where the transfer styles in the examples are the same

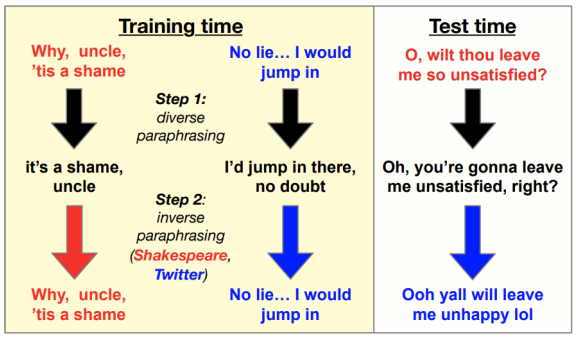

Figure 3: Two step text style transformation using fine-tuned language models. Borrowed from Krishna et al. (2020).

as the desired transfer style; and augmented zero-shot, where the transfer styles in the examples are varied and not just binary sentiment. They use a proprietary large language model and GPT-3 to test their methods. For evaluation, they use both automatic and human evaluation on parallel benchmarks. They also toy with fun styles where users ask arbitrary edit requests.

Finally, Suzgun et al. (2022) investigate prompt-based style transfer on smaller language models. Their method first generates several candidates in the target style based on the use of zero-shot or few-shot prompts of Reif et al. (2022). They then re-rank the candidates according to a combination of desired properties and pick the highest scorer.

3 Methods

3.1 GPT-3.5

We use OpenAI's ChatGPT. Specifically, we use `gpt-3.5-turbo`, which is the GPT-3.5 family's most capable and cost-effective model. We were able to get API access and run experiments at a minimal cost. As of the submission of this paper, the newest GPT model by OpenAI, GPT-4, is still difficult to use accessibly for experiments, as it has a higher cost per token generation, a more restrictive rate limit, and a wait list for API access[4]. From here on, we will refer to the model we use as GPT-3.5.

While GPT-3.5 follows the same architecture as GPT-3, its has better performance in many tasks thanks to reinforcement learning from human feedback (RLHF) (Ouyang et al., 2022). GPT-3 was available to and used by Reif et al. (2022) in experiments. However, as far as we are aware, GPT-3.5 has yet to be benchmarked on several popular text-style transfer tasks. Since GPT-3.5 is now an even stronger prompt-based language model, we find it reasonable to test its abilities on standard text-style transfer datasets.

We also use a frozen paraphraser by Krishna et al. (2020) which is GPT2-large model fine-tuned to paraphrase in diverse ways.

3.2 Prompting Methods

We propose two methods to test prompt-based text style transfer on GPT-3.5.

3.2.1 Straight Prompting using Zero-Shot and Augmented Zero-Shot

Following prompts designed by Reif et al. (2022) and reused in Suzgun et al. (2022), we test zero-shot and augmented zero-shot text style transfer on GPT-3.5. We found that their zero-shot prompt, left as it was, proved sufficiently simple and performed the same when trying a few example inputs as ablation. We also left the full text of the augmented zero-shot prompt untouched to see if it would perform as well without changes.

Both the zero-shot prompt and augmented zero-shot prompt use a starting delimiter to indicate the beginning of the text generation (ex: " Generate a sentence: ' "). While this was likely meant to clue earlier language models to end their generation with a closing delimiter, in our experience with using GPT-3.5, opening delimiters are no longer as necessary to use to still see interesting and desired output. Our prediction is that RLHF and longer training led GPT-3.5 to better identify the separation between instructions and the start of its generation, as well as deciding when to end. For this reason, we slightly modify the zero-shot and augmented zero-shot prompts by removing the opening delimiter. The straight prompts for Zero-Shot and Augmented Zero-Shot are shown in Figure 4.

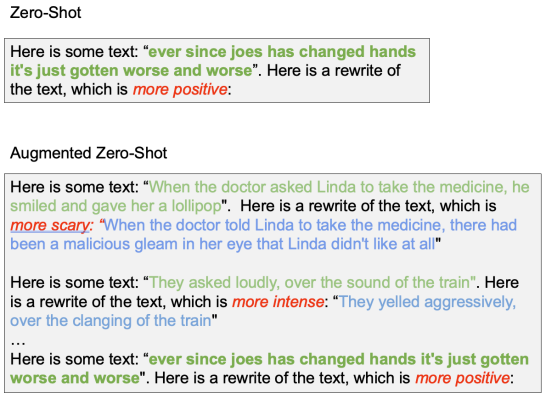

Figure 4: Straight Prompting using Zero-Shot and Augmented Zero-Shot for text style transfer. The boldface green text is the text we want to transform, the green texts are the texts we want to transform and the blue text are the texts transformed, the red text is the text style we expected.

3.2.2 Two step with paraphrasing and inverse paraphrasing

We propose using two-step prompting on large language models to improve performance on text transfer tasks, as in Figure 2. We treat GPT-3.5 as a drop-in replacement for the paraphrase and inverse-paraphraser in Krishna et al. (2020). First, we prompt GPT-3.5 to paraphrase and normalize an input text. Then, we use the paraphrased text as input to a fresh reset of the model (a new conversation) and use the straight zero-shot prompt. Before testing on the complete dataset, we experiment with a variety of constructed prompts for the paraphrase step. We want to remove style while preserving content and keep other changes minimal, like text length or vocabulary use.

To isolate testing GPT-3.5's ability as an inverse paraphraser, we also try using the paraphraser model from Krishna et al. (2020) and passing the paraphrased text as input to GPT-3.5 with the straight zero-shot prompt.

[4] https://platform.openai.com/docs/models

3.2.3 Baseline

Following the practice in Krishna et al. (2020), we also test a baseline method, COPY, that copies the original text as the output. We would expect the style transfer score to be extremely low, the similarity score to be high, and the fluency score to measure the fluency of the original texts.

4 Evaluating style transfer

4.1 Automatic v.s. Human

Both automatic and human evaluation methods exist for text style transfer. In automatic methods, the metrics are easily reproducible and can be calculated automatically given a parallel dataset where there are premade human references for possible translations. In human evaluation, the methods require manual checking or ranking of model outputs by annotators. As automatic evaluations are thought to be insufficient at comprehensively evaluating text generation quality (Liu et al., 2016; Novikova et al., 2017), many papers use both automatic evaluation and human evaluation (Luo et al., 2019; Reif et al., 2022; Luo et al., 2023; Roy et al., 2023; Mir et al., 2019; Suzgun et al., 2022; Krishna et al., 2020). However, for this project, we only use automatic evaluation and do not consider human or crowd-sourced evaluation because of the time limit and the human resource limit.

4.2 Criteria

To measure the success of text style transfer, we look for the generated text to score highly on three axes based on the proposal of Jin et al. (2022): transfer style strength, semantic preservation, and fluency. We discuss each property and a corresponding, appropriate, automated method for measurement.

4.3 Transfer Style Strength

To measure if the generated text has the target style, we use a frozen classifier model that is trained to predict if a text is one style or another. Then, the transfer style strength, or accuracy (ACC), is the percentage of generated texts that lead to correct classifications.

For the Yelp task where the binary classes are "negative" or "positive", we want a sentiment classifier. We follow the practice in Luo et al. (2023); Reif et al. (2022); Lai et al. (2021) and use SiEBERT (Hartmann et al., 2023), a fine-tuned RoBERTa-Large (Liu et al., 2019) for sentiment classification.

For the GYAFC task where the binary classes are "formal" or "informal", we want a formality classifier. We follow the practice in Luo et al. (2023); Reif et al. (2022); Lai et al. (2021) and use a fine-tuned RoBERTa-Large (Liu et al., 2019) for formality classification. While previous work fine-tuned their own formality classifiers, we find that the most popular formality classifier[5] on HuggingFace Transformers (Wolf et al., 2020) works as expected for our baseline evaluations.

4.4 Semantic Similarity

Semantic similarity is the shared content of the original and generated texts. A sentence that scores highly on style transfer score should still be relevant and connected to the original sentence by the vocabulary used. To measure content similarity, we use premade human-written references as a proxy and measure the overlap of the generated outputs with the reference style transfers.

There are two popular automated methods that compare the semantic similarity of a generated text to premade human references. The **Bi**lingual **E**valuation **U**nderstudy (BLEU) metric (Papineni et al., 2002) calculates precision: the number of overlapping tokens divided by the number of total tokens in the generated text. The Recall-Oriented Understudy for Gisting Evaluation (ROUGE) metric (Lin, 2004) calculates recall: the number of overlapping tokens divided by the number of total tokens in the reference text.

The simplest form of BLEU is the quotient of the matching words under the total count of words in the hypothesis sentence (traduction). Regarding the denominator BLEU is a precision-oriented metric.

$$\mathrm{p}_n = \frac{\sum_{m \in hypothesis} \mathrm{Count}_{\mathrm{Match}}(m)}{\sum_{m \in hypothesis} \mathrm{Count}(m)},$$

While there are more than 100 variants[6] of ROUGE, we think the classical ROUGE-N, which is based on n-grams matching between the reference and candidate summaries is the most commonly applied one. The ROUGE-N recall score is as follows:

$$\text{ROUGE-N}_R = \frac{\sum_{m \in \mathcal{M}_n} \mathrm{Count}_{\mathrm{Match}}(m)}{\sum_{m \in \mathcal{M}_n} \mathrm{Count}(m)},$$

[5] https://huggingface.co/s-nlp/roberta-base-formality-ranker

[6] https://aclanthology.org/D15-1013.pdf

where $\mathcal{M}_n$ is the set of unique n-grams in the reference summary, $\text{Count}(m)$ is the number of occurrence of n-gram m in the reference summary, $\text{Count}_{\text{Match}}(m)$ is the number of co-occurrence of n-gram m in both the reference summary and candidate summary.

BLEU was originally created to score translation tasks while ROUGE was originally created for summarization evaluation. While both could theoretically be used to measure semantic similarity in style transfer, in practice, BLEU is the most widely used (Jin et al., 2022).

The specific method of calculating BLEU has also been somewhat standardized. Most papers use `multi-bleu.perl`, which influenced the implementation of the sacreBLEU package (Post, 2018). SacreBLEU and previous, practical implementations of BLEU scorers use a brevity penalty that encourages shorter sentences. Else, a sentence could hack the metric by including many tokens to increase overlap with the original sentence. The Brevity Penalty (BP) is calculated as follows:

$$BP = \begin{cases} 1 & if l_{hyp} > l_{ref} \\ e^{1-\frac{l_{ref}}{l_{hyp}}} & if l_{hyp} \leq l_{ref} \end{cases}$$

which then leads to the final BLEU score:

$$BLEU_N = BP \cdot \exp\left(\sum_{n=1}^{N} w_n \log p_n\right)$$

4.5 Fluency

We measure the fluency of texts using average token-level perplexity, the exponent of the loss obtained by the model. We follow the practice in Suzgun et al. (2022); Luo et al. (2023); Reif et al. (2022), we calculate the perplexity of a token as measured by GPT2-Large (Radford et al., 2019) via HuggingFace Transformers (Wolf et al., 2020).

5 Experimental Setup

We evaluate our methods on Grammarly's Yahoo Answers Formality Corpus (GYAFC) (Rao and Tetreault, 2018) and Yelp Reviews (Zhang et al., 2016) using the evaluation methodology proposed in Section 4. We choose GYAFC and Yelp as they are used by several recent works, as seen in Table 1. With a test split size of 1332 and 1000 entries, respectively, the datasets are more accessible to test and experiment on than some other, larger benchmarks (Jin et al., 2022).

5.1 Dataset

The GYAFC corpus was created using the Yahoo Answers corpus "L6 - Yahoo! Answers Comprehensive Questions and Answers version 1.0"[7]. Sentences that contain URLs, are questions, and are shorter than 5 words or longer than 25 were removed from the original data. Based on the formality classifier from PT16 (Pavlick and Tetreault, 2016), "Entertainment & Music" and "Family & Relationships" which contain the most informal sentences are selected to create the GYAFC dataset (Rao and Tetreault, 2018). After careful analysis, we think GYAFC is of high quality. Moreover, many papers also include GYAFC as their benchmark (Krishna et al., 2020; Suzgun et al., 2022). For the metric evaluation, we choose the "Family & Relationships" part since many other papers only choose this part (Krishna et al., 2020; Suzgun et al., 2022).

Access to the GYAFC dataset requires users to first gain access to the Yahoo Answers corpus and then contact Joel Tetreault. The approval of getting access to the Yahoo Answers corpus and the correspondence with Joel Tetreault is shown in A.

The Yelp dataset is a collection of positive and negative reviews from Yelp reviews to train and evaluate text style transfer. The Yelp Reviews dataset was originally created for sentiment classification tasks. Yelp was extended to the text style transfer task after the addition of human annotations that translated each original text to a rewritten version in its opposite sentiment. The reference style transfers we use are from (Luo et al., 2019). Their annotations classify the rating above 3 stars as positive and below 3 as negative with the train, dev, and test split the same as Li et al. (2018). Since there are both positive and negative entries, Yelp provides two tasks: transferring a positive tone to a negative text and transferring a negative tone to a positive text.

5.2 Dataset Quality

Though the Yelp sentiment style transfer dataset is used by Luo et al. (2019); Reif et al. (2022), the quality of this dataset is not that high. The manually translated sentiment sentences might not make sense. We notice that for some negative sentences like "this place is awful!" and "that is ridiculous", their corresponding manually generated positive

[7] https://webscope.sandbox.yahoo.com/catalog.php?datatype=l&did=11

Paper	Models	Params	GYAFC?	YELP?
1	STRAP	774M	✓	
2	GPT-2-XL	1.6B	✓	✓
2	GPT-Neo-1.3B	3B	✓	
2	GPT-J-6B	6B		✓
3	LaMDA	137B		✓
3	GPT-3	175B		✓

Table 1: Summary of Models, Benchmarks

1. Style Transfer via Paraphrasing (STRAP): Reformulating Unsupervised Style Transfer as Paraphrase Generation (Krishna et al., 2020)
2. GPT-2-XL,GPT-J-6B,GPT-Neo-1.3B: Prompt-and-Rerank: A Method for Zero-Shot and Few-Shot Arbitrary Textual Style Transfer with Small Language Models (Suzgun et al., 2022)
3. GPT-3,LaMDA: A Recipe for Arbitrary Text Style Transfer with Large Language Models (Reif et al., 2022)

sentence is "gre, fun, knowledgable for a wonderfulat"[8]. We don't think of "GRE" as fun and this sentence also doesn't make sense to us. Moreover, for some positive sentences like "they are honest, professional, and quick." the corresponding manually generated negative sentence is "bor bored bored bored boreded". We hypothesize that the annotator felt bored when assigned this transformation task.

6 Result

We show our results on Yelp and GYAFC in Table 2 and Table 3, respectively.

6.1 Transfer Style Strength

For Yelp, GPT-3.5 using augmented zero-shot performs the highest in style transfer accuracy, setting a new state of the art at 94% and beating previous performances by at least 4 points. For GYAFC, we observe that all our methods score near perfect accuracy, with only one output classified as informal or no incorrect classifications at all. This indicates that GPT-3.5 has a strong handle on transferring formality into text, which seems reasonable if we consider that in RLHF, formal text and grammatically correct sentences are more highly rewarded.

The GPT paraphraser with a GPT inverse paraphraser performs the same or better than the frozen paraphraser with a GPT inverse paraphraser. Both methods perform worse than straight zero-shot or augmented zero-shot. For the GPT paraphraser, this indicates that perhaps more work is required to find a prompt that paraphrases and removes style from the text in the most ideal way. With the frozen paraphraser, which was trained on a smaller amount of data, out-of-distribution inputs are probably handled less robustly than GPT-3.5.

We also need to notice that for the naive copy method, the accuracy for $\text{Yelp}_{N->P}$ is 11%, $\text{Yelp}_{P->N}$ is 1%, GYAFC is 13%. That means either our sentiment classification model is not perfect or the data is slightly ambiguous for sentiment.

6.2 Semantic Similarity and Fluency

GPT-3.5 scores lower on the similarity metric, BLEU, and scores worse on the fluency score, perplexity (PPL). This is likely due to the longer outputs of GPT-3.5 as well as the lack of specification in the zero-shot or augmented zero-shot to keep the prompt short or semantically similar to the original text.

From qualitative observations, the BLEU scores are low because, while the generated texts have high semantic similarity with the original texts, they still differed with the choices made by human annotators. In Yelp, as discussed in 5.2, the differences are due to annotation quality. In GYAFC, for transferring the formality of "*The rest is easy from there!!*", GPT-3.5 straight zero-shot outputs "*It becomes simple after that point.*" while the references are *"From there, the rest is easy!"*, *"The remainder is simple from this point forward!"*, *"The rest becomes easy after this."*, and *"The rest is easy from there."*. The meaning is similar but the BLEU score is equal to 0.

[8]The references can be found at https://github.com/luofuli/DualRL/blob/master/data/yelp/tsf_template

Dataset	YELP		
Model	ACC	BLEU	PPL (GPT-2)
[1] DualRL*†	85.9	55.1	982
[1] DualRL**	88	25.9	133
[2] Prompt and Rerank(GPT-2-XL)**$_{P->N}$	87	14.8	65
[2] Prompt and Rerank(GPT-J-6B)**$_{P->N}$	87	23.0	80
[2] Prompt and Rerank(GPT-2-XL)**$_{N->P}$	72	12.0	**55**
[2] Prompt and Rerank(GPT-J-6B)**$_{N->P}$	65	20.2	58
[3] Lambada zero-shot*†	90.6	10.4	79
[3] Lambada five-shot*†	83.2	19.8	240
[3] Lambada augmented zero-shot*†	79.6	16.1	173
GPT-3.5 zero-shot*$_{P->N}$	91	6.2	108
GPT-3.5 augmented zero-shot*$_{P->N}$	**94**	5.2	96
Frozen Paraphraser, GPT Inverse*$_{P->N}$	87	2.2	100
GPT Paraphraser, GPT Inverse*$_{P->N}$	87	3.3	102
GPT-3.5 zero-shot*$_{N->P}$	71	4.3	65
GPT-3.5 augmented zero-shot*$_{N->P}$	86	4.3	68
Frozen Paraphraser, GPT Inverse*$_{N->P}$	76	2.0	67
GPT Paraphraser, GPT Inverse*$_{N->P}$	87	3.3	102
Baseline: copy the input as the output*$_{P->N}$	1	55.0	251
Baseline: copy the input as the output*$_{N->P}$	11	**56.7**	361

Table 2: Results on Yelp.

* : For these models, the performance is based on the Yelp sentiment style transfer dataset provided by Luo et al. (2019).
†: The data is provided by Reif et al. (2022)
** : For these models, the performance is based on the Yelp Restaurant Review provided by Zhang et al. (2016). The data are provided by Suzgun et al. (2022)
References: [1] (Luo et al., 2019), [2] (Suzgun et al., 2022), [3] (Reif et al., 2022)

Dataset	GYAFC		
Model	ACC	BLEU	PPL (GPT-2)
[1] DualRL*	71.1	41.9	-
[2] Prompt and Rerank (GPT-Neo-1.3B)**	85	36.4	68
[2] Prompt and Rerank (GPT-2-XL)**	82	32.7	**58**
GPT-3.5 zero-shot	99.9	8.8	65
GPT-3.5 augmented zero-shot	**100**	8.8	81
GPT paraphraser, GPT inverse paraphrase	100	5.2	88
Frozen Paraphraser, GPT Inverse Paraphraser	100	5.7	82
Baseline: copy the input as the output	14	51.0	199

Table 3: Results on GYAFC.

* : The data is provided by Luo et al. (2019)
** : The data are provided by Suzgun et al. (2022)
References: [1] (Luo et al., 2019), [2] (Suzgun et al., 2022)

7 Conclusion

In this project, we applied prompt-based and two-step methods using GPT-3.5 to achieve near-perfect scores for standard text-style translation tasks. While GPT-3.5 does not necessarily score high on the similarity or fluency scores, its ability to set the new state of the art on accuracy suggests that GPT-3.5 and similar models must be tested on more challenging style transfer tasks. As the power of general large language models exceeds the testing power of automatic evaluations, studying their abilities, such as in emulating author styles, becomes a more manually involved process that involves fields other than solely natural language processing.

Acknowledgements

We want to thank Professor Arman Cohan for his guidance and useful feedback on the direction of this project, Simeng Han and Yilun Zhao for showing sharing useful practices for calling the GPT API, and Joel Tetreault for providing us with the GYAFC dataset. Finally, we'd like to thank our classmates in CPSC 670, who have inspired us with their insights into so many interesting natural language processing papers.

References

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Jochen Hartmann, Mark Heitmann, Christian Siebert, and Christina Schamp. 2023. More than a feeling: Accuracy and application of sentiment analysis. *International Journal of Research in Marketing*, 40(1):75–87.

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

Touseef Iqbal and Shaima Qureshi. 2022. The survey: Text generation models in deep learning. *Journal of King Saud University - Computer and Information Sciences*, 34(6, Part A):2515–2528.

Di Jin, Zhijing Jin, Zhiting Hu, Olga Vechtomova, and Rada Mihalcea. 2022. Deep Learning for Text Style Transfer: A Survey. *Computational Linguistics*, 48(1):155–205.

Kalpesh Krishna, John Wieting, and Mohit Iyyer. 2020. Reformulating unsupervised style transfer as paraphrase generation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 737–762, Online. Association for Computational Linguistics.

Huiyuan Lai, Antonio Toral, and Malvina Nissim. 2021. Thank you bart! rewarding pre-trained models improves formality style transfer.

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

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Chia-Wei Liu, Ryan Lowe, Iulian Serban, Mike Noseworthy, Laurent Charlin, and Joelle Pineau. 2016. How NOT to evaluate your dialogue system: An empirical study of unsupervised evaluation metrics for dialogue response generation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2122–2132, Austin, Texas. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach.

Fuli Luo, Peng Li, Jie Zhou, Pengcheng Yang, Baobao Chang, Zhifang Sui, and Xu Sun. 2019. A dual reinforcement learning framework for unsupervised text style transfer. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence, IJCAI 2019*.

Guoqing Luo, Yu Tong Han, Lili Mou, and Mauajama Firdaus. 2023. Prompt-based editing for text style transfer.

Remi Mir, Bjarke Felbo, Nick Obradovich, and Iyad Rahwan. 2019. Evaluating style transfer for text.

Lili Mou and Olga Vechtomova. 2020. Stylized text generation: Approaches and applications. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: Tutorial Abstracts*, pages 19–22, Online. Association for Computational Linguistics.

Jekaterina Novikova, Ondřej Dušek, Amanda Cercas Curry, and Verena Rieser. 2017. Why we need new evaluation metrics for NLG. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2241–2252, Copenhagen, Denmark. Association for Computational Linguistics.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Ellie Pavlick and Joel Tetreault. 2016. An empirical analysis of formality in online communication. *Transactions of the Association for Computational Linguistics*, 4:61–74.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Brussels, Belgium. Association for Computational Linguistics.

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

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. 2022. A recipe for arbitrary text style transfer with large language models. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 837–848, Dublin, Ireland. Association for Computational Linguistics.

Shamik Roy, Raphael Shu, Nikolaos Pappas, Elman Mansimov, Yi Zhang, Saab Mansour, and Dan Roth. 2023. Conversation style transfer using few-shot learning.

Mirac Suzgun, Luke Melas-Kyriazi, and Dan Jurafsky. 2022. Prompt-and-rerank: A method for zero-shot and few-shot arbitrary textual style transfer with small language models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 2195–2222, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi 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 M. Rush. 2020. Huggingface's transformers: State-of-the-art natural language processing.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2016. Character-level convolutional networks for text classification.

A Appendix

Figure 5 and 6 are emails to get access to Yahoo Reviews and GYAFC.

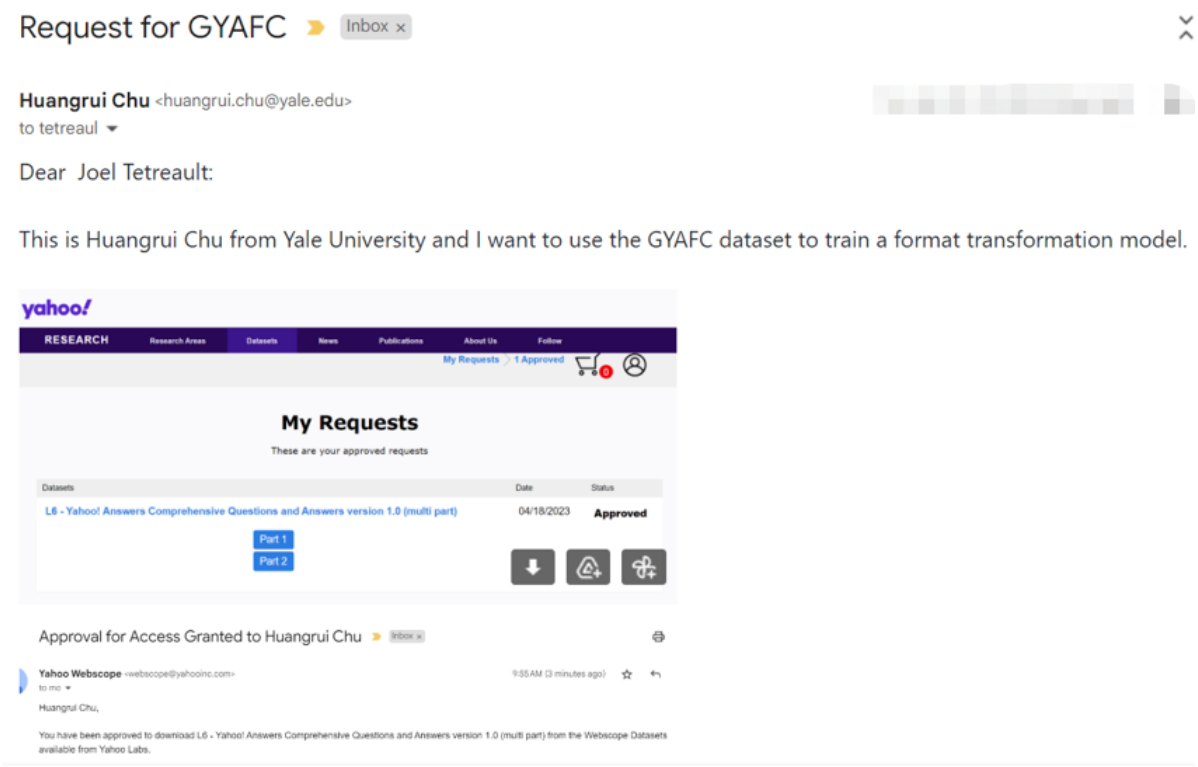

Figure 5: Approvement of YAHOO

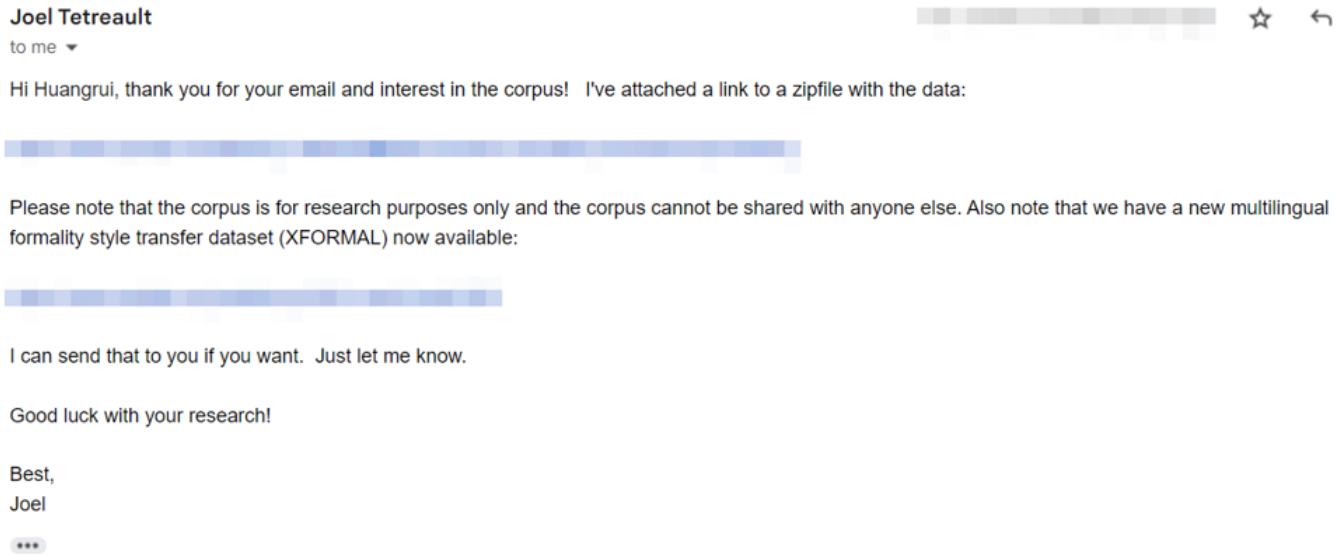

Figure 6: Access to GYAFC