CS230

Generative Text Style Transfer for Improved Language Sophistication

Robert Schmidt
rschm@stanford.edu

Spencer Braun
spencerb@stanford.edu

1 Introduction

Recent advances in Natural Language Processing (NLP) have heightened interest in generative text models and style transfer tasks. While most research has focused on binary sentiment transfer, some recent works like Jain et al. (2019)[1] and Rao et al. (2018)[2] have focused on text formality, a style that is less readily defined by specific key words. In that vein, we consider the problem of generative text style transfer to increase language sophistication, with the objective of rewriting an input sentence to preserve its meaning while improving its sophistication to match a target style text.

While early progress in the field focused on settings where parallel text is available, such as the Shakespeare style model developed in Jhamtani et al. (2017) [3], most practical settings benefit from non-parallel style transfer. For example, one noteworthy implementation is increasing writing sophistication for non-native English speakers. This holds value both in an educational setting as well as in the corporate world, where customer service text could be transferred to more standard English. We explore promising autoencoder and transformer architectures to achieve this goal.

2 Dataset and Preprocessing

Since there is no one notion of "style" in the field of NLP, most authors have defaulted to sentiment, utilizing the Yelp and IMDb reviews corpora [4]. However, no directly analogous dataset exists for distinguishing unrefined and more sophisticated texts. In order to leverage existing work, we have split our data into a set of "naive" and "sophisticated" sentences, such that the style is binary and highly differentiated between the two groups.

For the naive dataset, we searched for writing authored by or intended for students who were still acquiring their language skills. The dataset released for the Hewlett Foundation's Kaggle competition "Automated Student Assessment Prize (ASAP)" [1], which features anonymized graded essays authored by students in grades 7 through 10, fit our mission closely and made up the bulk of our unsophisticated dataset. These essays were supplemented by some simple, informal essays published on the site My Kids Way [2].

The sophisticated dataset was harder to compose, as we sought well-structured writing with a rich vocabulary without significant idiosyncratic style. Ultimately, a number of popular texts on Project Gutenberg [3] were concatenated along with modern texts from the Oxford Text Archive [4].

2.1 Preprocessing

All datasets were run through a number of preprocessing steps before entering a model. Texts were taken and split into sentences, discarding those with fewer than 40 characters, a number of uncommon punctuation marks, or all capital letters. The ASAP dataset contained a large number of misspellings; in order to train a model with a standard English vocabulary, these sentences were removed from consideration. The remaining sentences were again filtered to fewer than 35 words in order to limit the size of embeddings loaded into memory.

[1] The Hewlett Foundation: Automated Essay Scoring; https://www.kaggle.com/c/asap-aes/

[2] https://www.mykidsway.com/essays/

[3] https://www.gutenberg.org/

[4] https://ota.bodleian.ox.ac.uk/repository/xmlui/

2.2 Challenges and Dataset Modifications

Because sentence sophistication is not a simple to measure binary classification, modified datasets produced quite different results across model architectures. Therefore, the composition of the dataset was treated as an additional hyperparameter, necessary to tune in order to properly alter the features of the style transferred sentences. Data modifications fell into several categories:

- **Authorship**: While most of the sophisticated texts were selected from top books downloaded on Gutenberg, some authors had styles too distinct for this task. For example, Charles Dickens' work was included in some datasets, but removed later once models became fixated on his style.
- **Punctuation**: Sophisticated texts used diverse and complex syntax; its inclusion or exclusion tested whether it affected transferred content and fluency.
- **Proper Nouns**: The ASAP dataset was anonymized and retained only tokens instead of proper nouns. In some cases those tokens were removed, while in others they were standardized to be present in student and sophisticated texts. Similarly, the sophisticated texts contained a number of unique proper nouns; in some instances, those nouns were replaced by standardized tokens.

Ultimately, 3 different sophisticated training corpora were created, allowing us to see how changing each feature altered the style transfer task. Corpus 1 selected works in philosophy, education, and literature from popular texts on Project Gutenberg. Corpus 2 removed texts with distinctive proper nouns and added additional modern writing from the Oxford Text Archive. Finally, Corpus 3 used similar texts but replaced proper nouns and numbers with tags used by Stanford's Named Entity Recognition Tagger [5]. The full list of texts included in each sophisticated writing corpus can be found at `https://github.com/spencerbraun/sophisticated_style_transfer`.

3 Architecture Search

3.1 Initial Model Exploration

Cross-Aligned Autoencoder: Our architecture search began with the seminal work of Shen et al. (2017) [6], who proposed a cross-aligned autoencoder, so named because it "directly aligns the transferred samples from one style with the true samples from the other." While the model's emphasis on sentence reconstruction loss and generative framework informed our search going forward, we soon turned towards more recent research for our implementation.

Disentangled Style Transfer: John et al. (2019) [7] proposed a novel method of disentangling the style and content of input before training, such that the training procedure [10] can better recognize the difference between the two distributions. The complexity of this procedure is matched by its computational demands, and hence we decided to delve further in our architecture search.

3.2 Adversarially Regularized Autoencoder

The first model we found success with was the adversarially regularized autoencoder (ARAE) developed by Zhao et al. (2018) [8]. In the sentiment transfer setting, the authors were able to improve upon the performance of the cross-aligned Shen et al. (2017) model. Zhao et al. empirically learn a variable encoding and prior, which is ideal for our non-parallel style transfer task. More specifically, the authors employ LSTMs for their encoder/decoder network.

Loss function: The model loss is a weighted sum of reconstruction loss, classification loss, and Wasserstein distance. Each loss plays an important role in controlling training: reconstruction loss ensures that the model is able to properly ensure content preservation, while classification loss adversarially pushes the network to produce sentences that sufficiently match the target style. Moreover, the Wasserstein distance (difference between input and latent distributions) ensures the true and model distributions do not diverge significantly. The loss is explicitly specified in equation (1):

$$\mathcal{L} = \mathcal{L}_{rec}(\phi, \psi) + \lambda^{(1)} W(\mathbb{P}_Q, \mathbb{P}_{\mathbf{z}}) - \mathcal{L}_{class}(\phi, u) \tag{1}$$

Training procedure: At each training iteration, the ARAE first trains the encoder/decoder for reconstruction. Then, it trains a "critic" discriminator to distinguish between real and generated samples, while ensuring that the prior encodes all relevant information except the style. Finally, the ARAE trains the encoder/decoder adversarially, again ensuring that the style is kept separate from the encoding procedure.

3.3 Style Transformer: Transfer without Disentangled Representation

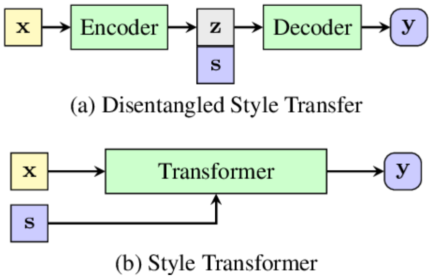
While the ARAE competes with Shen et al.'s cross-alignment procedure, the Dai et al. (2019) style transformer [4] answers the disentangled approach of John et al. [9] Instead, Dai et al. employ an architecture that does not assume a fixed latent representation and utilizes attention in a feed-forward network for both the encoder and decoder. [12] The difference between the authors' approach and the disentangled approach is summarized in Figure 1, where $\mathbf{z}$ is the style-independent content vector and $\mathbf{s}$ is the relevant style variable.

Loss function: The loss function is as follows, with parameters ϕ for the discriminator and θ for the style transformer:

$$\mathcal{L} = \mathcal{L}_{disc}(\phi) + \lambda^{(1)}\mathcal{L}_{self}(\theta) + \lambda^{(2)}\mathcal{L}_{cycle}(\theta) + \lambda^{(3)}\mathcal{L}_{style}(\theta) \tag{2}$$

Given an input $\mathbf{x}$, the transformer tries to model a function $\mathbf{y} = f_\theta(\mathbf{x}, \mathbf{s})$ which either maps an input to its style-transferred output $\hat{\mathbf{y}}$ (for target style $\mathbf{s} = \hat{\mathbf{s}}$) or reconstructs the original sentence. Since we employ a multi-class discriminator, the discriminator d_ϕ also attempts to distinguish generated data from real data.

- $\mathcal{L}_{disc}(\phi)$: discriminator cross-entropy loss for classifying real vs. generated sentences
- $\mathcal{L}_{self}(\theta)$: negative-log likelihood loss for reconstructing input $\mathbf{x}$ given model output $\mathbf{y}$
- $\mathcal{L}_{cycle}(\theta)$: negative-log likelihood loss for reconstructing $\mathbf{x}$ given style-transformed output $\hat{\mathbf{y}}$
- $\mathcal{L}_{style}(\theta)$: negative log-likelihood related to correctly classifying $\hat{\mathbf{y}}$ as $\hat{\mathbf{s}}$



(a) Disentangled Style Transfer

(b) Style Transformer

Figure 1: Disentangled vs. transformer [[4] Fig. 1]

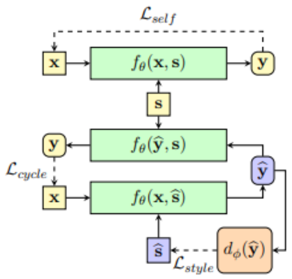

$\mathcal{L}_{self}$
x
$f_\theta(\mathbf{x},\mathbf{s})$
y
s
y
$f_\theta(\hat{\mathbf{y}},\mathbf{s})$
$\hat{\mathbf{y}}$
$\mathcal{L}_{cycle}$
x
$f_\theta(\mathbf{x},\hat{\mathbf{s}})$
$\hat{\mathbf{s}}$
$d_\phi(\hat{\mathbf{y}})$
$\mathcal{L}_{style}$

Figure 2: Transformer training process [[4] Fig. 2]

4 Hyperparameter Search

ARAE: A variety of hyperparameters were modified in the ARAE, but they seemed to have little effect on the overall outcome. The main tested changes involved changing the dropout rate, increasing the training iterations for the GAN generators, and altering the composition of the dataset trained on. All generated indistinguishable output, so more attention was paid to other promising models instead of widening the hyperparameter search.

Style Transformer: Generally, we found the most success with the authors' [12] preset hyperparameters; namely, we utilized the preset weight factors of $\lambda^{(1)} = 0.25$, $\lambda^{(2)} = 0.5$, and $\lambda^{(3)} = 1$ for training. While Dai et al. employed a slower learning rate of $\alpha = 0.0001$ for both the discriminator and style-transformer networks, we also attempted a run with $\alpha = 0.001$. However, this was to the detriment of our output, which suffered generally lower scores across our array of evaluation metrics.

We also tried different batch sizes, although mainly as a computational necessity. The model default size of 64 achieved comparable performance to training on batches of size 32. Lastly, we tried experimenting with the depth of the network, increasing from the default 4 to a 6-layer network. As will be touched upon in our following section, deeper networks had a positive effect on our results.

5 Evaluation

5.1 Methodology

There is no one succinct metric that can summarize the sophistication of a sentence; however, viewing style transfer through a holistic lens still allows a quantitative picture to form for trial evaluation. We built on conventions established by Jain et al. (2019) and Zhao et al. (2018) [1][8], grading our transferred sentences on content preservation, approximation of the target style, and overall fluency.

- **Content Preservation**: The BLEU score was the primary metric used to determine how much a transferred sentence retained of its original content. For this task, an ideal BLEU score would be somewhere in the middle of its range, with values close to 1 failing to learn enough style differences and those close to 0 far removed from the content the original sentence. Cosine similarity was also considered as a preservation metric using GloVe embeddings [9], but ultimately showed little variation and was less useful in distinguishing among the results.
- **Style Approximation**: We trained a 5-gram language model on our target style dataset using KenLM [10], allowing us to calculate the perplexity score (PPL) of generated sentences. Sentences with low perplexity scores were more likely to come from the same distribution as the target text. Additionally, the PINC score, introduced by Chen et al. (2011) [11], uses n-gram comparisons to measure the novelty of the transferred sentence. It acts as a counterweight to BLEU, and also ideally falls towards the middle of its range.
- **Fluency**: The Flesch Kincaid readability tests are designed to measure how easily a passage could be understood [1]. We used the Flesch Reading Ease scale and the Flesch-Kincaid Grade Level tests on our transferred sentences to measure their fluency and adherence to standard English syntax.

5.2 Results

Training Corpus	Model Settings	BLEU	PINC	PPL	F-K Ease	F-K Grade
Corpus 1		0.432	0.528	1044.24	81.97	5.60
Corpus 1	LR = 0.001	0.228	0.730	1638.63	67.08	9.10
Corpus 2		0.361	0.626	629.92	80.62	6.00
Corpus 2	NP	0.224	0.751	616.94	62.01	10.70
Corpus 3		0.381	0.587	282.45	75.20	6.80
Corpus 3	NP, GloVe	0.097	0.838	73.09	103.63	2.50
Corpus 3	GloVe	0.074	0.843	53.36	88.74	3.70
Corpus 3	Deep	0.514	0.434	390.92	72.16	6.40

Table 1: Evaluation metrics on best-performing iterations of the Style Transformer model. The corpus modifier "LR" refers to the learning rate, "GloVe" refers to pre-trained embeddings, and "NP" indicates punctuation was removed from both reference and target style training datasets. BLEU, Cosine, and PINC take average scores across the test dataset, while F-K Ease and F-K Grade are the median scores.

The autoencoder approach showed the least promise in distinguishing styles and maintaining content between reference and target texts. Specifically, the ARAE model struggled to make progress in training, as the model depends on a more well-defined corpus of contextual words. While vocabulary plays a role in differentiating our amateur and sophisticated texts, under a number of different hyperparameter choices the ARAE failed to learn to the differences in the texts, leading to mostly unintelligible output.

On the other hand, the Style Transformer showed much more promise in learning the differences in reference and target texts and exhibited significant sensitivity to the training texts and hyperparameters employed. Table 1 provides a quantitative summary of the variation in metrics over different iterations of the model. While qualitative judgment is also useful in determining the success of the trials, some clear trends are present in the evaluation data.

Corpus 1	Reference: Transferred:	therefore, he should prepare notes regularly at home. therefore, he wopsle, prepare notes regularly at home.
Corpus 3	Reference: Transferred:	it was dead silent . it was dead silent , and the of the of the of the of the .

Table 2: Example reference sentences and style-transferred output from the Style-Transformer model. Training on Corpus 1, the model focused on proper nouns present in the sophisticated text (here "Wopsle," a Dickens character). Replacing those proper nouns with tokens to form Corpus 3 produced different aberrant behavior.

It is important to note that dataset and hyperparameter tuning was an iterative process closely attuned to issues exhibited by a given model's output. For example, Corpus 1 retained proper nouns such as locations and character names in works of literature. When the trained model was applied to the test reference data, many style-transferred sentences contained references to these proper nouns as the model closely associated this unique vocabulary with its learned understanding of sophistication.

In Corpus 3, all proper nouns were replaced with tokens common to the amateur and sophisticated texts. This change eliminated the issues of Corpus 1, but introduced novel misunderstandings of sophistication. Since many of the sophisticated sentences were longer and more heavily punctuated, models trained on Corpus 3 appended words with high probabilities to the ends of sentences to improve their style scores. Table 2 contains example sentences for both issues described.

The choice of embedding matrix had a large impact on the output as well; models either made use of embeddings trained by the model or pre-trained GloVe embeddings. The models that utilized pre-trained embeddings skewed too far towards learning the target style while failing to retain the content of the reference text, resulting in very low BLEU scores while achieving the lowest PPL scores of any model. Due to the large model sizes and computational constraints, more recent language models such as BERT were not tested but serve as an area for future research.

5.3 Summary

Overall, the Style Transformer performed best with Corpus 3 without pre-trained embeddings, and keeping punctuation. Models with these training parameters tended to maintain moderate BLEU scores while reducing PPL significantly from models trained on other corpora. Additionally, increasing the number of layers from 4 to 6 for the transformer (Corpus 3, Deep) showed promise in producing sentences that retained content while exhibiting a clear change in stylistic vocabulary.

Corpus 1	Reference: Transferred:	so we say something. so we say steerforth!
Corpus 2	Reference: Transferred:	in conclusion all these elements of the desert affect the cyclist in some way. in conclusion all these elements of the called portrait the savage in some way. the in the
Corpus 3, Deep	Reference: Transferred:	all relationships need a little laughter to lighten the mood once in awhile . all relationships blind a little physics to lighten the hypothesis once in awhile .

Table 3: Comparison of reference texts to outputs from different iterations of style transfer models.

6 Conclusion

Our analysis indicates that the notion of content in sentiment transfer is insufficient for the unsupervised sophistication transfer task. While outputs in our best model preserve sentence form, vocabulary changes radically alter the implicit content of the sentences; in other words, we observe that the best models are preserving appearance, but not necessarily meaning. With further syntactical research, we could define more suitable losses and evaluation metrics that better direct our non-parallel sophistication transfer.

Overall, our results demonstrate an optimistic future for sophistication transfer while also highlighting a number of challenges that must be considered. We see further avenues for improvement in developing attention mechanisms for grammar, incorporating more recent language models like BERT and GPT-2, and employing parallel text supplementary networks to improve understanding of idiomatic phrases. Many of these improvements require substantial computational resources, so additional exploration into how pre-trained and off-the-shelf models could be substituted for different network components is worthwhile as well. In short, while our model encountered significant obstacles, our findings open up many promising deep learning approaches to non-parallel text sophistication.

7 Contributions

Both team members contributed equally to this project. Initially, Robert focused on literature review and model comparisons while Spencer focused on data collection and processing. Both team members then contributed to model training, hyperparameter search, and evaluation.

References

[1] P. Jain, A. Mishra, A. P. Azad, and K. Sankaranarayanan, “Unsupervised controllable text formalization,” *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, p. 6554–6561, Jul 2019.

[2] S. Rao and J. Tetreault, “Dear sir or madam, may i introduce the gyafc dataset: Corpus, benchmarks and metrics for formality style transfer,” *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, 2018.

[3] H. Jhamtani, V. Gangal, E. Hovy, and E. Nyberg, “Shakespearizing modern language using copy-enriched sequence to sequence models,” *Proceedings of the Workshop on Stylistic Variation*, 2017.

[4] N. Dai, J. Liang, X. Qiu, and X. Huang, “Style transformer: Unpaired text style transfer without disentangled latent representation,” *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019.

[5] J. R. Finkel, T. Grenager, and C. Manning, “Incorporating non-local information into information extraction systems by Gibbs sampling,” in *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL’05)*, (Ann Arbor, Michigan), pp. 363–370, Association for Computational Linguistics, June 2005.

[6] T. Shen, T. Lei, R. Barzilay, and T. Jaakkola, “Style transfer from non-parallel text by cross-alignment,” 2017.

[7] V. John, L. Mou, H. Bahuleyan, and O. Vechtomova, “Disentangled representation learning for non-parallel text style transfer,” *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019.

[8] J. Zhao, Y. Kim, K. Zhang, A. M. Rush, and Y. LeCun, “Adversarially regularized autoencoders,” 2017.

[9] J. Pennington, R. Socher, and C. D. Manning, “Glove: Global vectors for word representation,” in *Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1532–1543, 2014.

[10] K. Heafield, “KenLM: Faster and smaller language model queries,” in *Proceedings of the Sixth Workshop on Statistical Machine Translation*, (Edinburgh, Scotland), pp. 187–197, Association for Computational Linguistics, July 2011.

[11] D. L. Chen and W. B. Dolan, “Collecting highly parallel data for paraphrase evaluation,” in *ACL*, 2011.

[12] A. Joulin, E. Grave, P. Bojanowski, and T. Mikolov, “Bag of tricks for efficient text classification,” *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, 2017.

[13] R. Y. Pang and K. Gimpel, “Unsupervised evaluation metrics and learning criteria for non-parallel textual transfer,” 2018.

[14] W. Xu, A. Ritter, B. Dolan, R. Grishman, and C. Cherry, “Paraphrasing for style,” in *COLING*, pp. 2899–2914, 2012.

[15] K. Carlson, A. Riddell, and D. Rockmore, “Evaluating prose style transfer with the bible,” *Royal Society Open Science*, vol. 5, p. 171920, Oct 2018.

[16] T. Kim, M. Cha, H. Kim, J. K. Lee, and J. Kim, “Learning to discover cross-domain relations with generative adversarial networks,” 2017.

[17] Z. Fu, X. Tan, N. Peng, D. Zhao, and R. Yan, “Style transfer in text: Exploration and evaluation,” 2017.

[18] W. Xu, C. Callison-Burch, and C. Napoles, “Problems in current text simplification research: New data can help,” *Transactions of the Association for Computational Linguistics*, vol. 3, pp. 283–297, 2015.

[19] J. Lee, Z. Xie, C. Wang, M. Drach, D. Jurafsky, and A. Ng, “Neural text style transfer via denoising and reranking,” in *Proceedings of the Workshop on Methods for Optimizing and Evaluating Neural Language Generation*, (Minneapolis, Minnesota), pp. 74–81, Association for Computational Linguistics, June 2019.

[20] D. Liu and G. Liu, “A transformer-based variational autoencoder for sentence generation,” *2019 International Joint Conference on Neural Networks (IJCNN)*, 2019.