Reformulating Unsupervised Style Transfer as Paraphrase Generation

Kalpesh Krishna[♠] **John Wieting**[♢] **Mohit Iyyer**[♠]

[♠]University of Massachusetts Amherst, [♢]Carnegie Mellon University
{kalpesh,miyyer}@cs.umass.edu
jwieting@cs.cmu.edu

Project Page: http://style.cs.umass.edu

Abstract

Modern NLP defines the task of *style transfer* as modifying the style of a given sentence without appreciably changing its semantics, which implies that the outputs of style transfer systems should be paraphrases of their inputs. However, many existing systems purportedly designed for style transfer inherently warp the input's meaning through *attribute transfer*, which changes semantic properties such as sentiment. In this paper, we reformulate unsupervised style transfer as a paraphrase generation problem, and present a simple methodology based on fine-tuning pretrained language models on automatically generated paraphrase data. Despite its simplicity, our method significantly outperforms state-of-the-art style transfer systems on both human and automatic evaluations. We also survey 23 style transfer papers and discover that existing automatic metrics can be easily gamed and propose fixed variants. Finally, we pivot to a more real-world style transfer setting by collecting a large dataset of 15M sentences in 11 diverse styles, which we use for an in-depth analysis of our system.

1 Introduction

The task of *style transfer* on text data involves changing the style of a given sentence while preserving its semantics.[1] Recent work in this area conflates style transfer with the related task of *attribute transfer* (Subramanian et al., 2019; He et al., 2020), in which modifications to attribute-specific content words (e.g., those that carry sentiment) warp both stylistic *and* semantic properties of a sentence (Preotiuc-Pietro et al., 2016). Attribute transfer has been criticized for its limited real-world applications: Pang (2019) argue that semantic preservation is critical for author obfuscation (Shetty et al., 2018), data augmentation (Xie et al., 2019; Kaushik et al., 2020), text simplification (Xu et al., 2015), writing assistance (Heidorn, 2000). Moreover, semantic preservation (via paraphrases) has several applications like better translation evaluation (Sellam et al., 2020; Freitag et al., 2020) and adversarial defenses (Iyyer et al., 2018).

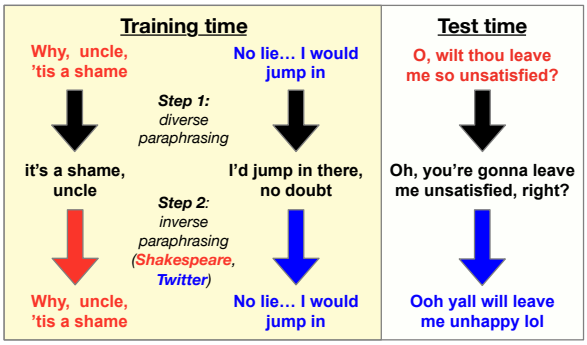

Figure 1: During training, STRAP applies a diverse paraphraser to an input sentence and passes the result through a style-specific *inverse paraphraser* to reconstruct the input. At test time, we perform style transfer by swapping out different inverse paraphrase models (Shakespeare → Twitter shown here). All generated sentences shown here are **actual outputs from STRAP**.

We propose to improve semantic preservation in style transfer by modeling the task as a controlled paraphrase generation problem. Our unsupervised method (**S**tyle **Tra**nsfer via **P**araphrasing, or STRAP) requires no parallel data between different styles and proceeds in three simple stages:

1. Create pseudo-parallel data by feeding sentences from different styles through a diverse paraphrase model (Figure 1, left).
2. Train style-specific *inverse* paraphrase models that convert these paraphrased sentences back into the original stylized sentences.
3. Use the inverse paraphraser for a desired style to perform style transfer (Figure 1, right).

[1]We use the *quasi-paraphrase* definition of semantic equivalence from Bhagat and Hovy (2013) throughout this paper. We loosely define *style* as patterns in lexical and syntactic choice within the space of quasi-paraphrases.

Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, pages 737–762,
November 16–20, 2020. ©2020 Association for Computational Linguistics

Our approach requires none of the finicky[2] modeling paradigms popular in style transfer research — no reinforcement learning (Luo et al., 2019), variational inference (He et al., 2020), or autoregressive sampling during training (Subramanian et al., 2019). Instead, we implement the first two stages of our pipeline by simply fine-tuning a pretrained GPT-2 language model (Radford et al., 2019).

Despite its simplicity, STRAP significantly outperforms the state of the art on formality transfer and Shakespeare author imitation datasets by **2-3x** on automatic evaluations and **4-5x** on human evaluations. We further show that only 3 out of 23 prior style transfer papers properly evaluate their models: in fact, a naïve baseline that randomly chooses to either copy its input or retrieve a random sentence written in the target style *outperforms* prior work on poorly-designed metrics.

Finally, we take a step towards real-world style transfer by collecting a large dataset CDS (**C**orpus of **D**iverse **S**tyles) of 15M English sentences spanning **11 diverse styles**, including the works of James Joyce, romantic poetry, tweets, and conversational speech. CDS is orders of magnitude larger and more complex than prior benchmarks, which generally focus on transferring between just two styles. We analyze STRAP's abilities on CDS, and will release it as a benchmark for future research.

In summary, **our contributions** are:

(1) a simple approach to perform lexically and syntactically diverse paraphrasing with pretrained language models;

(2) a simple unsupervised style transfer method that models semantic preservation with our paraphraser and significantly outperforms prior work;

(3) a critique of existing style transfer evaluation based on a naïve baseline that performs on par with prior work on poorly designed metrics;

(4) a new benchmark dataset that contains 15M sentences from 11 diverse styles.

2 Style Transfer via Paraphrasing

We loosely define *style* as common patterns of lexical choice and syntactic constructions that are distinct from the content of a sentence, following prior work (Hovy, 1987; DiMarco and Hirst, 1993; Green and DiMarco, 1993; Kabbara and Cheung, 2016). While we acknowledge this distinction is not universally accepted,[3] this treatment is critical to unlock several real-world applications of style transfer (as argued in Section 1). Unfortunately, many modern style transfer systems do not respect this definition: a human evaluation (Table 2) shows that **fewer than 25% of style-transferred sentences** from two state-of-the-art systems (Subramanian et al., 2019; He et al., 2020) on formality transfer were rated as paraphrases of their inputs.

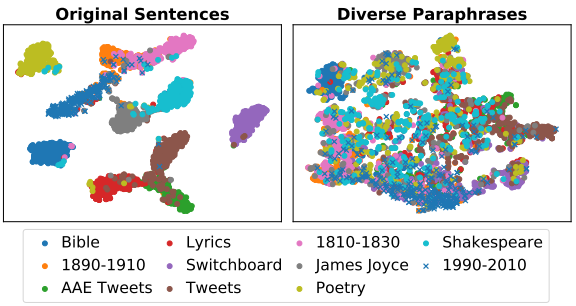

Figure 2: **Diverse paraphrasing normalizes sentences by removing stylistic identifiers.** We cluster validation sentences from our CDS dataset by applying t-SNE to `[CLS]` vectors from a RoBERTa style classifier. The original sentences (left) form distinct clusters, while the paraphrased sentences (right) do not, showing the stylized text has been normalized.

Motivated by this result, we reformulate style transfer as a controlled paraphrase generation task. We call our method **STRAP**, or **S**tyle **Tra**nsfer via **P**araphrasing. STRAP operates within an unsupervised setting: we have raw text from distinct target styles, but no access to parallel sentences paraphrased into different styles. To get around this lack of data, we create *pseudo-parallel* sentence pairs using a paraphrase model (Section 2.1) trained to maximize output diversity (Section 2.4). Intuitively, this paraphrasing step normalizes the input sentence by stripping away information that is predictive of its original style (Figure 2). The normalization effect allows us to train an *inverse paraphrase* model specific to the original style, which attempts to generate the original sentence given its normalized version (Section 2.2). Through this process, the model learns to identify and produce salient features of the original style without unduly warping the input semantics.

2.1 Creating pseudo-parallel training data

The first stage of our approach involves normalizing input sentences by feeding them through a

[2]For example, reproducing deep RL methods is challenging (Henderson et al., 2018), vanilla adversarial training is unstable (Arjovsky et al., 2017), and VAEs suffer from posterior collapse (Bowman et al., 2016).

[3]For example, Eckert (2008) considers style and semantics to be inseparable; while Meyerhoff (2015) considers style to be intra-speaker variation in different social contexts

diverse paraphrase model. Consider a corpus of sentences from multiple styles, where the set of all sentences from style i is denoted by $\mathbf{X}^i$. We first generate a paraphrase $\mathbf{z}$ for every sentence $\mathbf{x} \in \mathbf{X}^i$ using a pretrained paraphrase model f_{para},

$$\mathbf{z} = f_{\text{para}}(\mathbf{x}) \text{ where } \mathbf{x} \in \mathbf{X}^i.$$

This process results in a dataset $\mathbf{Z}^i$ of normalized sentences and allows us to form a pseudo-parallel corpus $(\mathbf{X}^i, \mathbf{Z}^i)$ between each original sentence and its paraphrased version. Figure 2 shows that this paraphrasing process has a powerful style normalization effect for our instantiation of f_{para}.

2.2 Style transfer via inverse paraphrasing

We use this pseudo-parallel corpus to train a style-specific model that attempts to reconstruct the original sentence $\mathbf{x}$ given its paraphrase $\mathbf{z}$. Since f_{para} removes style identifiers from its input, the intuition behind this *inverse* paraphrase model is that it learns to insert stylistic features through the reconstruction process. Formally, the inverse paraphrase model f^i_{inv} for style i learns to reconstruct[4] the original corpus $\mathbf{X}^i$ using the standard language modeling objective with cross-entropy loss $\mathcal{L}_{\text{CE}}$,

$$\bar{\mathbf{x}} = f^i_{\text{inv}}(\mathbf{z}) \text{ where } \mathbf{z} \in \mathbf{Z}^i$$
$$\text{loss} = \sum_{\mathbf{x} \in \mathbf{X}^i} \mathcal{L}_{\text{CE}}(\mathbf{x}, \bar{\mathbf{x}})$$

During inference, given an arbitrary sentence $\mathbf{s}$ (in any particular style), we convert it to a sentence $\bar{\mathbf{s}}^j$ in target style j using a two-step process of style normalization with f_{para} followed by stylization with the inverse paraphraser f^j_{inv}, as in

$$\bar{\mathbf{s}}^j = f^j_{\text{inv}}(f_{\text{para}}(\mathbf{s})).$$

2.3 Paraphraser implementation with GPT-2

We fine-tune the large-scale pretrained GPT2-large language model (Radford et al., 2019) to implement both the paraphraser f_{para} and inverse paraphrasers f^i_{inv} for each style.[5] Starting from a pretrained LM improves both output fluency and generalization to small style-specific datasets (Section 5). We use the encoder-free seq2seq modeling approach described in Wolf et al. (2018), where input and output sequences are concatenated together with a separator token. We use Hugging Face's Transformers library (Wolf et al., 2019) to implement our models; see Appendix A.2 for more details about the architecture & hyperparameters.

2.4 Promoting diversity by filtering data

The final piece to our approach is how we choose training data for the paraphrase model f_{para}. We discover that maximizing lexical and syntactic diversity of the output paraphrases is crucial for effective style normalization (Section 5, 6). We promote output diversity by training f_{para} on an aggressively-filtered subset of PARANMT-50M (Wieting and Gimpel, 2018), a large corpus of backtranslated text. Specifically, we apply three filters: (1) removing sentence pairs with more than 50% trigram or unigram overlap to maximize lexical diversity and discourage copying; (2) removing pairs with lower than 50% reordering of shared words, measured by Kendall's tau (Kendall, 1938), to promote syntactic diversity; and (3) removing pairs with low semantic similarity, measured by the SIM model from Wieting et al. (2019).[6] After applying these filters, our training data size shrinks from 50M to 75K sentence pairs, which are used to fine-tune GPT-2; see Appendix A.1 for more details about the filtering process and its effect on corpus size.

3 Evaluating style transfer

Providing a meaningful comparison of our approach to existing style transfer systems is difficult because of (1) poorly-defined automatic and human methods for measuring style transfer quality (Pang, 2019; Mir et al., 2019; Tikhonov et al., 2019), and (2) misleading (or absent) methods of aggregating three individual metrics (transfer accuracy, semantic similarity and fluency) into a single number. In this section, we describe the flaws in existing metrics and their aggregation (the latter illustrated through a naïve baseline), and we propose a new evaluation methodology to fix these issues.

3.1 Current state of style transfer evaluation

We conduct a survey of **23** previously-published style transfer papers (more details in Appendix A.9), which reveals three common

[4] This process resembles **denoising autoencoders** (Vincent et al., 2008; Lample et al., 2018, DAE): f_{para} acts as a semantic preserving noise function; f^i_{inv} reconstructs the input.

[5] We fine-tune a separate GPT-2 model f^i_{inv} per style. Section 5 shows that this outperforms a single inverse paraphraser shared across all styles with style input.

[6] This model achieves strong performance on semantic textual similarity (STS) SemEval benchmarks (Agirre et al., 2016). We remove all pairs with a score lower than 0.7.

properties on which style transfer systems are evaluated. Here, we discuss how prior work implements evaluations for each of these properties and propose improved implementations to address some of their downsides.

Transfer accuracy (ACC): Given an output sentence $\bar{\mathbf{s}}^j$ and a target style j, a common way of measuring transfer success is to train a classifier to identify the style of a transferred sentence and report its accuracy ACC on generated sentences (i.e., whether $\bar{\mathbf{s}}^j$ has a predicted style of j). 14 of 23 surveyed papers implement this style classifier with a 1-layer CNN (Kim, 2014). However, recent large Transformers like BERT (Devlin et al., 2019) significantly outperform CNNs on most NLP tasks, including style classification. Thus, we build our style classifier by fine-tuning RoBERTa-large (Liu et al., 2019) on all our datasets, leading to significantly more reliable ACC evaluation.[7]

Semantic similarity (SIM): A style transfer system can achieve high ACC scores without maintaining the semantics of the input sentence, which motivates also measuring how much a transferred sentence deviates in meaning from the input. 15 / 23 surveyed papers use n-gram metrics like BLEU (Papineni et al., 2002) against reference sentences, often along with self-BLEU with the input, to evaluate semantic similarity. Using BLEU in this way has many problems, including (1) unreliable correlations between n-gram overlap and human evaluations of semantic similarity (Callison-Burch et al., 2006), (2) discouraging output diversity (Wieting et al., 2019), and (3) not upweighting important semantic words over other words (Wieting et al., 2019; Wang et al., 2020). These issues motivate us to measure semantic similarity using the subword embedding-based SIM model of Wieting et al. (2019), which performs well on semantic textual similarity (STS) benchmarks in SemEval workshops (Agirre et al., 2016).[8]

Fluency (FL): A system that produces ungrammatical outputs can still achieve high scores on both ACC and SIM, motivating a separate measure for fluency. Only 10 out of 23 surveyed papers did a fluency evaluation; 9 of which used language model perplexity, which is a poor measure because (1) it is unbounded and (2) unnatural sentences with common words tend to have low perplexity (Mir et al., 2019; Pang, 2019). To tackle this we replace perplexity with the accuracy of a RoBERTa-large classifier trained on the CoLA corpus (Warstadt et al., 2019), which contains sentences paired with grammatical acceptability judgments. In Table 1, we show that our classifier marks most reference sentences as fluent, confirming its validity.[9]

Human evaluation: As automatic evaluations are insufficient for evaluating text generation (Liu et al., 2016; Novikova et al., 2017), 17 out of 23 surveyed style transfer papers also conduct human evaluation. In our work, we evaluate SIM and FL using human evaluations.[10] As we treat style transfer as a paraphrase generation task, we borrow the three-point scale used previously to evaluate paraphrases (Kok and Brockett, 2010; Iyyer et al., 2018), which jointly captures SIM and FL. Given the original sentence and the transferred sentence, annotators on Amazon Mechanical Turk can choose one of three options: **0** for no paraphrase relationship; **1** for an ungrammatical paraphrase; and **2** for a grammatical paraphrase. A total of 150 sentence pairs were annotated per model, with three annotators per pair. More details on our setup, payment & agreement are provided in Appendix A.10.

3.2 Aggregation of Metrics

So far, we have focused on individual implementations of ACC, SIM, and FL. After computing these metrics, it is useful to aggregate them into a single number to compare the overall style transfer quality across systems (Pang, 2019). However, only 5 out of the 23 papers aggregate these metrics, either at the corpus level (Xu et al., 2018; Pang and Gimpel, 2019) or sentence level (Li et al., 2018). Even worse, the corpus-level aggregation scheme can be easily gamed. Here, we describe a naïve system that outperforms state-of-the-art style transfer systems when evaluated using corpus-level aggregation, and we present a new sentence-level aggregation metric that fixes the issue.

The issue with corpus-level aggregation: Aggregating ACC, SIM, and FL is inherently difficult

[7] The RoBERTa style classifier, built with `fairseq` (Ott et al., 2019), achieves a test accuracy of 90.4% on the Shakespeare data(vs 83.5% for CNN) and 94.8% on the Formality data (vs 92.4%). The datasets are introduced in Section 4.1.

[8] For reference, we evaluate with BLEU in Appendix A.5.

[9] Mir et al. (2019) also recommended a similar method to evaluate fluency instead of perplexity, where they train classifiers to distinguish between machine / human sentences.

[10] We do not conduct human evaluations for ACC since style classification is difficult for an untrained crowdsourced worker unfamiliar with the set of target styles.

because they are inversely correlated with each other (Pang, 2019). Prior work has combined these three scores into a single number using geometric averaging (Xu et al., 2018) or learned weights (Pang and Gimpel, 2019). However, the aggregation is computed *after* averaging each metric independently across the test set (*corpus-level aggregation*), which is problematic since systems might generate sentences that optimize only a subset of metrics. For example, a Shakespeare style transfer system could output *Wherefore art thou Romeo?* regardless of its input and score high on ACC and FL, while a model that always copies its input would score well on SIM and FL (Pang, 2019).

A Naïve Style Transfer System: To concretely illustrate the problem, we design a naïve baseline that exactly copies its input with probability p and chooses a random sentence from the target style corpus for the remaining inputs, where p is tuned on the validation set.[11] When evaluated using geometric mean corpus-level aggregation (GM column of Table 1) this system *outperforms* state of the art methods (UNMT, DSLM) on the Formality dataset despite not doing any style transfer at all!

Proposed Metric: A good style transfer system should **jointly** optimize all metrics. The strong performance of the naïve baseline with corpus-level aggregation indicates that metrics should be combined at the sentence level *before* averaging them across the test set (*sentence aggregation*). Unfortunately, **only 3** out of 23 surveyed papers measure absolute performance after *sentence-level aggregation*, and all of them use the setup of Li et al. (2018), which is specific to human evaluation with Likert scales. We propose a more general alternative,

$$J(\text{ACC}, \text{SIM}, \text{FL}) = \sum_{x \in \mathbf{X}} \frac{\text{ACC}(x) \cdot \text{SIM}(x) \cdot \text{FL}(x)}{|\mathbf{X}|}$$

where x is a sentence from a test corpus $\mathbf{X}$. We treat ACC and FL at a sentence level as a binary judgement, ensuring incorrectly classified or disfluent sentences are automatically assigned a score of 0. As a sanity check, our naïve system performs extremely poorly on this new metric (Table 1), as input copying will almost always yield an ACC of zero, while random retrieval results in low SIM.

[11] p = 0.4 / 0.5 for Formality / Shakespeare datasets.

4 Experiments & Results

We evaluate our method (STRAP) on two existing style transfer datasets, using the evaluation methodology proposed in Section 3. Our system significantly outperforms state of the art methods and the naïve baseline discussed in Section 3.2.

4.1 Datasets

We focus exclusively on semantics-preserving style transfer tasks, which means that we do not evaluate on *attribute transfer* datasets such as sentiment, gender, and political transfer. Specifically, we use two standard benchmark datasets for Shakespeare author imitation and formality transfer to compare STRAP against prior work. While both datasets contain parallel data, we only use it to automatically evaluate our model outputs; for training, we follow prior work by using the non-parallel train-validation-test splits from He et al. (2020).

The **Shakespeare author imitation** dataset (Xu et al., 2012) contains 37k training sentences from two styles — William Shakespeare's original plays, and their modernized versions. Shakespeare's plays are written in Early Modern English, which has a significantly different lexical (e.g., *thou* instead of *you*) and syntactic distribution compared to modern English. Our second dataset is **Formality transfer** (Rao and Tetreault, 2018), which contains 105k sentences, also from two styles. Sentences are written either in formal or informal modern English. Unlike formal sentences, informal sentences tend to have more misspellings, short forms (*u* instead of *you*), and non-standard usage of punctuation.

4.2 Comparisons against prior work

We compare STRAP on the Shakespeare / Formality datasets against the following baselines:

- COPY: a lower bound that simply copies its input, which has been previously used in prior work (Subramanian et al., 2019; Pang, 2019)
- NAÏVE: our method from Section 3.2 that randomly either copies its input or retrieves a sentence from the target style
- REF: an upper bound computed by evaluating reference sentences using our metrics
- UNMT: unsupervised neural machine translation from Subramanian et al. (2019)
- DLSM: the deep latent sequence model from

Model	Formality (GYAFC)					Shakespeare				
	ACC	SIM	FL	GM(A,S,F)	J(A,S,F)	ACC	SIM	FL	GM(A,S,F)	J(A,S,F)
COPY	5.2	80.1	88.4	33.3	4.2	9.6	67.1	79.1	37.1	7.2
NAÏVE	58.9	38.9	89.1	58.9	7.3	49.9	34.9	78.9	51.6	4.1
REF	93.3	100	89.7	94.2	83.8	90.4	100	79.1	89.4	70.5
UNMT (2019)	**78.5**	49.1	52.5	58.7	20.0	70.5	37.5	49.6	50.8	14.6
DLSM (2020)	78.0	47.7	53.7	58.5	18.6	71.1	43.5	49.4	53.5	16.3
STRAP ($p = 0.0$)	67.7	**72.5**	**90.4**	**76.3**	**45.5**	71.7	**56.4**	**85.2**	**70.1**	**34.7**
STRAP ($p = 0.6$)	70.7	69.9	88.5	75.9	44.5	75.7	53.7	82.7	69.5	33.5
STRAP ($p = 0.9$)	76.8	62.9	77.4	72.0	38.3	**79.8**	47.6	71.7	64.8	27.5

Table 1: Automatic evaluation of our method STRAP (using different p values for nucleus sampling) against prior state-of-the-art methods (UNMT, DLSM), lower bound baselines (COPY, NAÏVE) and reference sentences (REF). STRAP significantly outperforms prior work, especially on our proposed $J(\cdot)$ metric. GM is the geometric mean.

Dataset	Model	ACC	SIM	J(A,S)	J(A,S,F)
Form.	UNMT	77.3	22.7	14.7	7.3
	DLSM	78.0	24.0	15.3	10.0
	$p = 0.0$	71.3	**76.0**	**54.7**	**41.3**
	$p = 0.9$	**79.3**	56.7	46.0	28.0
Shak.	UNMT	69.3	20.7	10.0	7.3
	DLSM	65.3	37.3	21.3	9.3
	$p = 0.0$	70.7	**79.3**	**56.0**	**47.3**
	$p = 0.9$	**74.7**	54.0	38.0	24.7

Table 2: Human evaluation of STRAP with greedy decoding ($p = 0.0$) and nucleus sampling ($p = 0.9$) shows large improvements (**4-5x**) on both the Formality (Form.) and Shakespeare (Shak.) datasets. Details on metric calculations are provided in Appendix A.10.

Dataset	Model	ACC	SIM	FL	J(A,S,F)
Form.	STRAP	67.7	72.5	90.4	45.5
	– Inf. PP	27.5	78.5	88.2	20.7
	– Mult. PP	63.1	72.0	90.8	42.3
	– Div. PP	61.2	79.5	88.7	43.8
	– GPT2	84.6	43.8	61.7	23.1
	GPT2-md	71.0	70.7	88.6	45.8
	GPT2-sm	69.1	68.6	87.6	42.9
Shak.	STRAP	71.7	56.4	85.2	34.7
	– Inf. PP	40.1	66.1	76.3	23.3
	– Mult. PP	45.9	56.5	91.1	24.8
	– Div. PP	49.7	64.4	82.9	28.2
	– GPT2	75.6	26.7	66.9	13.6
	GPT2-md	73.4	54.0	86.4	34.3
	GPT2-sm	68.0	53.2	84.6	31.5

Table 3: Ablation study using automatic metrics on the Formality (Form.) and Shakespeare (Shak.) datasets.

He et al. (2020), which is currently state-of-the-art on both datasets.[12]

STRAP significantly outperforms the prior state of the art (DLSM) on automatic metrics (Table 1) with a $J(\cdot)$ score of 45.5 (vs 18.6) on Formality and 34.7 (vs 16.3) on Shakespeare. The improvements are even larger when SIM and FL are measured through human evaluations (Table 2): in this setting, STRAP achieves 41.3 (vs 10.0) on Formality and 47.3 (vs 9.3) on Shakespeare. Across the board, STRAP significantly improves in SIM and FL while maintaining similar ACC. Finally, the large gap between REF and STRAP on automatic metrics provides exciting avenues for future research.[13]

[12]We use the implementations of both UNMT and DLSM made publicly available by He et al. (2020), and we verify that their UNMT model performs on par with reported sentiment transfer numbers in Subramanian et al. (2019). The original code of Subramanian et al. (2019) has not been open-sourced.

[13]Results with other metrics such as BLEU, as well as comparisons against several other baselines like Li et al. (2018); Prabhumoye et al. (2018); Luo et al. (2019); Dai et al. (2019); Sudhakar et al. (2019) are provided in Appendix A.5. STRAP significantly outperforms all prior work.

5 Ablation studies

In this section, we perform several ablations on STRAP to understand which of its components contribute most to its improvements over baselines. Overall, these ablations validate the importance of both paraphrasing and pretraining for style transfer.

Paraphrase diversity improves ACC: How critical is diversity in the paraphrase generation step? While our implementation of f_{para} is trained on data that is heavily-filtered to promote diversity, we also build a non-diverse paraphrase model by removing this diversity filtering of PARANMT-50M but keeping all other experimental settings identical. In Table 3, the *–Div. PP* rows show a drop in ACC across both datasets as well as higher SIM, which in both cases results in a lower $J(\cdot)$ score. A qualitative inspection reveals that the decreased ACC and increased SIM are both due to a greater degree of input copying, which motivates the importance of diversity.

Paraphrasing during inference improves ACC:

The diverse paraphraser f_{para} is obviously crucial to train our model, as it creates pseudo-parallel data for training f^i_{inv}, but is it necessary during inference? We try directly feeding in the original sentence (without the initial paraphrasing step) to the inverse paraphrase model f^i_{inv} during inference, shown in the *–Inf. PP* row of Table 3. While SIM and FL are largely unaffected, there is a large drop in ACC, bringing down the overall score (45.5 to 20.7 in Formality, 34.7 to 23.3 in Shakespeare). This supports our hypothesis that the paraphrasing step is useful for normalizing the input.

LM pretraining is crucial for SIM and FL: As we mainly observe improvements on FL and SIM compared to prior work, a natural question is how well does STRAP perform without large-scale LM pretraining? We run an ablation study by replacing the GPT-2 implementations of f_{para} and f^i_{inv} with LSTM seq2seq models, which are trained with global attention (Luong et al., 2015) using OpenNMT (Klein et al., 2017) with mostly default hyperparameters.[14] As seen in the – *GPT2* row of Table 3, this model performs competitively with the UNMT / DLSM models on J(ACC,SIM,FL), which obtain 20.0 / 18.6 on Formality (Table 1), respectively. However, it is significantly worse than STRAP, with large drops in SIM and FL.[15] This result shows the merit of both our algorithm and the boost that LM pretraining provides.[16]

Nucleus sampling trades off ACC for SIM: While our best performing system uses a greedy decoding strategy, we experiment with nucleus sampling (Holtzman et al., 2020) by varying the nucleus p value in both Table 1 and Table 2. As expected, higher p improves diversity and trades off increased ACC for lowered SIM. We find that $p = 0.6$ is similar to greedy decoding on $J(\cdot)$ metrics, but higher p values degrade performance.

Multiple inverse paraphrasers perform better than a single style-conditional model: Finally, we explore a more parameter-efficient alternative to training a separate inverse paraphrase model per style. Prior work in conditioned language models feed style-specific tokens as input to a single model to control the output style (Johnson et al., 2017; Keskar et al., 2019). We adopt this strategy by feeding in the average RoBERTa-large `[CLS]` vectors (by running the pretrained classifier on the training data) as style indicators to a single f^i_{inv}. We present results in Table 3 as *–Mult. PP*. This strategy is not as effective as training a different paraphrase model per style, primarily due to drop in ACC. We hypothesize this is due to a conflict between different style distributions in a single model.

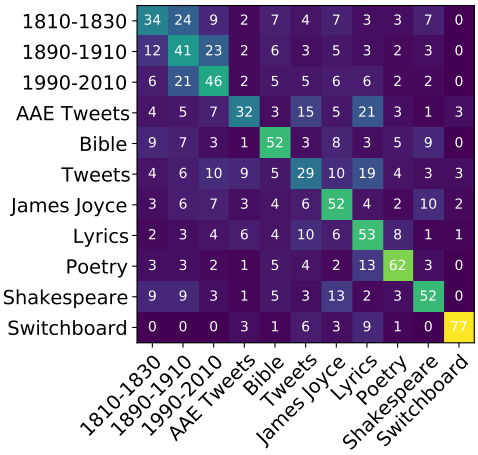

Figure 3: **Classifier confusion after style transfer**. Every row shows the classifier label distribution on sentences *transferred to* the target style (the row label). The off-diagonal elements show mis-classifications with intuitive domain similarities, such as (Lyrics, Poetry); (AAE, Tweets); (Joyce, Shakespeare).

6 Towards Real-World Style Transfer

All of our experiments and ablations thus far have been on the Shakespeare and Formality datasets, which contain just two styles each. To explore the ability of our system to perform style transfer between many diverse styles, we create the **C**orpus of **D**iverse **S**tyles (CDS), a new non-parallel style transfer benchmark dataset with 11 diverse styles (15M tokens), and use it to evaluate STRAP.

Corpus of Diverse Styles: To create CDS, we obtain data (Table 5) from existing academic research datasets (Godfrey et al., 1992; Blodgett et al., 2016) and public APIs or online collections like Project Gutenberg (Hart, 1992). We choose styles that are easy for human readers to identify at a sentence level (e.g., Tweets or Biblical text), and the left side of Figure 2 confirms that machines also cluster CDS into eleven distinct styles. While prior benchmarks involve a transfer between two styles, CDS has 110 potential transfer directions.

[14] The only hyperparameter we tune is the learning rate schedule. More details in Appendix A.4.

[15] A qualitative inspection of outputs confirms the LSTM struggles to maintain semantics. We suspect this is due to lack of training data ($<$ 75K pairs) to learn a powerful paraphraser.

[16] Additionally, we note that weaker pretrained language models like GPT2-medium (*GPT2-md*) perform similarly to GPT2-large, while GPT2-small (*GPT2-sm*) is notably worse.

Direction	Input	Output
Tweet → Shak.	yall kissing before marriage?	And you kiss'd before your nuptial?
Lyrics → AAE	It's a good thing you don't have bus fare	It's a good thing u aint gettin no ticket
Tweet → Poetry	Its so disrespectful I hate ittttttt	For 'tis so rude, I have a sick regard,
Shak. → Tweet	Need you my help?	Are yall okay? Like do you need my help??
AAE → Shak.	Aint got nooooo holiday spirit frfr	No spirit of this holiday, gentle man.
Poetry → Bible	For the foul thief is just at your gate.	the wicked thief art at thy door.
1990. → Tweet	Now I can't talk to him about it.	I dont wanna talk to him abt that
1990. → Poetry	I don't know why, but I sensed there was something wrong.	Something felt wrong; I knew not why,
Tweet → Bible	when ur going thru an episode and cant cope anymore	when thou art in an episode, and cannot stand it any more;
1890. → AAE	I was just thinking the same thing that you have expressed.	u said the same thing i thought lol
1990. → 1810.	He was being terrorized into making a statement by the same means as the other so-called "witnesses."	Terror had been employed in the same manner with the other witnesses, to compel him to make a declaration.
AAE → Shak.	If I got a dollar every time one of my friends told me they hate me, I'd be rich	I would have been rich, had I but a dollar for every friend that hath said they hate me.
Joyce → Bible	I appeal for clemency in the name of the most sacred word our vocal organs have ever been called upon	I beseech thee in the name of the most holy word which is in our lips, forgive us our trespasses.

Table 4: Example outputs from STRAP trained on our CDS dataset (more generations in Appendix A.11).

Style	Size	Style	Size
Shakespeare	27.5K	Lyrics	5.1M
James Joyce	41.2K	1810-1830	216.0K
English Tweets	5.2M	1890-1910	1.3M
AAE Tweets	732.3K	1990-2010	2.0M
Romantic Poetry	29.8K	Bible	34.8K
Switchboard	148.8K		

Table 5: List of styles in our dataset along with the their total sizes. The year periods (like "1810-1830") refer to sentences from the Corpus of Historical American English (Davies, 2012). "AAE Tweets" refers to African American English Tweets corpus from Blodgett et al. (2016). "Switchboard" is a collection of conversational speech transcripts from Godfrey et al. (1992). Details of the collection and examples are in Appendix A.6.

Shakespeare ↔ English Tweets, CDS				
Model	ACC	SIM	FL	J(A,S,F)
COPY	0.1	100.0	69.2	0.0
UNMT (2019)	76.7	20.6	37.7	4.4
DLSM (2020)	64.2	19.6	33.1	2.0
STRAP ($p = 0.0$)	20.3	65.0	81.1	8.7
STRAP ($p = 0.6$)	31.1	58.1	75.0	10.8
STRAP ($p = 0.9$)	43.2	54.5	68.3	**13.9**

Table 6: A controlled comparison between models on 2 styles from CDS using automatic evaluation. ACC is calculated using our **11-way** CDS classifier and SIM is with input. STRAP greatly outperforms prior work.

We present dataset examples, details on collection and style similarity analysis in Appendix A.6.

Diverse paraphrasing normalizes stylized text With eleven styles, we can better validate the effectiveness of our diverse paraphraser at normalizing input sentences. After training an 11-way style classifier on CDS using RoBERTa-large, we observe an accuracy of **88.9%** on the original validation set. After paraphrasing the validation set with f_{para}, this classifier only correctly classifies **42.5%** sentences, indicating a significant decrease in recognizable stylistic features. Figure 2 further demonstrates this normalization effect. Finally, the magnitude of normalization is lower with the non-diverse paraphraser (from Section 5), with a smaller accuracy drop to **51.5%** after paraphrasing;[17] qualitatively, the diverse model exhibits more lexical swaps and syntactic diversity.[18]

Style Transfer on CDS: We measure STRAP's performance on CDS using Section 3's evaluation methodology. We sample 1K sentences from each style and use STRAP to transfer these sentences to each of the 10 other styles. Despite having to deal with many more styles than before, our system achieves **48.4%** transfer accuracy (on a 11-way RoBERTa-large classifier), a paraphrase similarity score of **63.5**, and **71.1%** fluent generations, yielding a J(ACC,SIM,FL) score of 20.7. A break-

[17] Even if we retrain the classifiers on a paraphrased version of the training set (to model the distribution better), the performance is only 65.8% for the diverse model and 72.3% for the non-diverse model, indicating a loss in style signal.

[18] On average, the diverse model has 51% unigram F1 word overlap and 27% word shuffling measured by Kendall's τ_B, compared to 28% unigram F1 and 6% shuffling for the non-diverse model; Appendix A.7 has a style-wise breakdown.

Direction	Input → Paraphrase → Output	Analysis
Shak. → Bible	Have you importuned him by any means? → did you ever try to import him? → hast thou ever tried to import him?	Misunderstanding the word "importune" — the model believes it refers to *import* rather than *harass / bother*.
1990. → Tweet.	The machine itself is made of little straws of carbon. → the machine is made of straw. → Machine made of straw.	Dropping of important semantic words during diverse paraphrasing ("carbon") significantly warps the meaning of sentences
Swit. → Shak.	well they offer classes out at uh Ray Hubbard → they're offering a course at Ray Hubbard's. → They do offer a course at the house of the Dukedom.	Hallucination of tokens irrelevant to the input ("house of the dukedom") to better reflect style distribution.
Tweet → Swit.	Knoxville aint for me → I'm not in Knoxville. → i don't know Knoxville	Subtle modifications in semantics since the models fail to understand their inputs.

Table 7: Representative examples showing the common failure modes of STRAP when evaluated on CDS.

down of style-specific performance is provided in Appendix A.8. An error analysis shows that the classifier misclassifies some generations as styles sharing properties with the target style (Figure 3).

Controlled comparisons: To ground our CDS results in prior work, we compare STRAP with baselines from Section 4.2. We sample equal number of training sentences from two challenging styles in CDS (Shakespeare, English Tweets) and train all three models (UNMT, DLSM, STRAP) on this subset of CDS.[19] As seen in Table 6, STRAP greatly outperforms prior work, especially in SIM and FL. Qualitative inspection shows that baseline models often output arbitrary style-specific features, completely ignoring input semantics (explaining poor SIM but high ACC).

Qualitative Examples: Table 4 contains several outputs from STRAP; see Appendix A.11 for more examples. We also add more qualitative analysis of the common failures of our system in Table 7. Our model makes mistakes similar to contemporary text generation systems — poor understanding of rare words, dropping / modification of semantic content, hallucination to better reflect training distribution.

7 Related Work

Unsupervised style transfer is often modeled by disentangling style & content using attribute classifiers (Hu et al., 2017; Shen et al., 2017), policy gradient training (Xu et al., 2018; Luo et al., 2019) or retrieval-based approaches (Li et al., 2018). Recently, backtranslation has emerged as a method to model semantic preservation (Prabhumoye et al., 2018), but this method can also warp semantics as seen in Subramanian et al. (2019); as such, we only use it to build our paraphraser's training data after heavy filtering. Our work relates to recent efforts that use Transformers in style transfer (Sudhakar et al., 2019; Dai et al., 2019). Closely related to our work is Gröndahl and Asokan (2019), who over-generate paraphrases using a complex handcrafted pipeline and filter them using proximity to a target style corpus. Instead, we *automatically learn* style-specific paraphrasers and do not need over-generation at inference. Relatedly, Preotiuc-Pietro et al. (2016) present qualitative style transfer results with statistical MT paraphrasers. Other, less closely related work on control & diversity in text generation is discussed in Appendix A.12.

8 Conclusion

In this work we model style transfer as a controlled paraphrase generation task and present a simple unsupervised style transfer method using diverse paraphrasing. We critique current style transfer evaluation using a survey of 23 papers and propose fixes to common shortcomings. Finally, we collect a new dataset containing 15M sentences from 11 diverse styles. Possible future work includes (1) exploring other applications of diverse paraphrasing, such as data augmentation; (2) performing style transfer at a paragraph level; (3) performing style transfer for styles unseen during training, using few exemplars provided during inference.

Acknowledgements

We thank the anonymous reviewers and area chair for their useful comments. We are grateful to Su Lin Blodgett, Katie Keith, Brendan O'Connor, Tu Vu, Shufan Wang, Nader Akoury, Rajarshi Das, Andrew Drozdov, Rico Angell, and the rest of the

[19] We could not find an easy way to perform 11-way style transfer in the baseline models without significantly modifying their codebase / model due to the complex probabilistic formulation beyond 2 styles and separate modeling for each of the 110 directions.

UMass NLP group for help at various stages of the project. Finally, we thank Arka Sadhu, Richard Pang, Kevin Gimpel, Graham Neubig, Diyi Yang, Shrimai Prabhumoye, Junxian He and Yonatan Belinkov for insightful discussions. This research was supported in part by a research gift from Adobe.

References

Eneko Agirre, Carmen Banea, Daniel Cer, Mona Diab, Aitor Gonzalez-Agirre, Rada Mihalcea, German Rigau, and Janyce Wiebe. 2016. SemEval-2016 task 1: Semantic textual similarity, monolingual and cross-lingual evaluation. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*.

Nader Akoury, Kalpesh Krishna, and Mohit Iyyer. 2019. Syntactically supervised transformers for faster neural machine translation. In *Proceedings of the Association for Computational Linguistics*.

Martin Arjovsky, Soumith Chintala, and Léon Bottou. 2017. Wasserstein generative adversarial networks. In *Proceedings of the International Conference of Machine Learning*.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. In *Proceedings of the International Conference on Learning Representations*.

Rahul Bhagat and Eduard Hovy. 2013. Squibs: What is a paraphrase? *Computational Linguistics*, 39(3):463–472.

Su Lin Blodgett, Lisa Green, and Brendan O'Connor. 2016. Demographic dialectal variation in social media: A case study of African-American English. In *Proceedings of Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Ondřej Bojar, Ondřej Dušek, Tom Kocmi, Jindřich Libovický, Michal Novák, Martin Popel, Roman Sudarikov, and Dušan Variš. 2016. CzEng 1.6: Enlarged Czech-English Parallel Corpus with Processing Tools Dockered. In *Text, Speech, and Dialogue: 19th International Conference, TSD 2016*, number 9924 in Lecture Notes in Computer Science, pages 231–238, Cham / Heidelberg / New York / Dordrecht / London. Masaryk University, Springer International Publishing.

Samuel Bowman, Luke Vilnis, Oriol Vinyals, Andrew Dai, Rafal Jozefowicz, and Samy Bengio. 2016. Generating sentences from a continuous space. In *Conference on Computational Natural Language Learning*.

Chris Callison-Burch, Miles Osborne, and Philipp Koehn. 2006. Re-evaluating the role of Bleu in machine translation research. In *Proceedings of the European Chapter of the Association for Computational Linguistics*.

Mingda Chen, Qingming Tang, Sam Wiseman, and Kevin Gimpel. 2019. Controllable paraphrase generation with a syntactic exemplar. In *Proceedings of the Association for Computational Linguistics*, Florence, Italy. Association for Computational Linguistics.

Ning Dai, Jianze Liang, Xipeng Qiu, and Xuanjing Huang. 2019. Style transformer: Unpaired text style transfer without disentangled latent representation. In *Proceedings of the Association for Computational Linguistics*.

Sumanth Dathathri, Andrea Madotto, Janice Lan, Jane Hung, Eric Frank, Piero Molino, Jason Yosinski, and Rosanne Liu. 2020. Plug and play language models: a simple approach to controlled text generation. In *Proceedings of the International Conference on Learning Representations*.

Mark Davies. 2012. Expanding horizons in historical linguistics with the 400-million word corpus of historical american english. *Corpora*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Conference of the North American Chapter of the Association for Computational Linguistics*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Chrysanne DiMarco and Graeme Hirst. 1993. A computational theory of goal-directed style in syntax. *Computational Linguistics*, 19(3):451–499.

Penelope Eckert. 2008. Variation and the indexical field 1. *Journal of sociolinguistics*, 12(4).

Joseph L Fleiss. 1971. Measuring nominal scale agreement among many raters. *Psychological bulletin*, 76(5):378.

Markus Freitag, David Grangier, and Isaac Caswell. 2020. Bleu might be guilty but references are not innocent. *arXiv preprint arXiv:2004.06063*.

Zhenxin Fu, Xiaoye Tan, Nanyun Peng, Dongyan Zhao, and Rui Yan. 2018. Style transfer in text: Exploration and evaluation. In *Association for the Advancement of Artificial Intelligence*.

John J Godfrey, Edward C Holliman, and Jane McDaniel. 1992. Switchboard: Telephone speech corpus for research and development. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*.

Tanya Goyal and Greg Durrett. 2020. Neural syntactic preordering for controlled paraphrase generation. *Proceedings of the Association for Computational Linguistics*.

Stephen J Green and Chrysanne DiMarco. 1993. Stylistic decision-making in natural language generation. In *European Workshop on Trends in Natural Language Generation*, pages 125–143. Springer.

Tommi Gröndahl and N Asokan. 2019. Effective writing style imitation via combinatorial paraphrasing. *arXiv preprint arXiv:1905.13464*.

Ankush Gupta, Arvind Agarwal, Prawaan Singh, and Piyush Rai. 2018. A deep generative framework for paraphrase generation. In *Association for the Advancement of Artificial Intelligence*.

Kelvin Guu, Tatsunori B Hashimoto, Yonatan Oren, and Percy Liang. 2018. Generating sentences by editing prototypes. *Transactions of the Association for Computational Linguistics*.

Michael Hart. 1992. The history and philosophy of Project Gutenberg. *Project Gutenberg*.

Junxian He, Xinyi Wang, Graham Neubig, and Taylor Berg-Kirkpatrick. 2020. A probabilistic formulation of unsupervised text style transfer. In *Proceedings of the International Conference on Learning Representations*.

George Heidorn. 2000. Intelligent writing assistance. *Handbook of natural language processing*, pages 181–207.

Peter Henderson, Riashat Islam, Philip Bachman, Joelle Pineau, Doina Precup, and David Meger. 2018. Deep reinforcement learning that matters. In *Association for the Advancement of Artificial Intelligence*.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2020. The curious case of neural text degeneration. In *Proceedings of the International Conference on Learning Representations*.

Eduard Hovy. 1987. Generating natural language under pragmatic constraints. *Journal of Pragmatics*, 11(6):689–719.

Zhiting Hu, Zichao Yang, Xiaodan Liang, Ruslan Salakhutdinov, and Eric P Xing. 2017. Toward controlled generation of text. In *Proceedings of the International Conference of Machine Learning*.

Mohit Iyyer, John Wieting, Kevin Gimpel, and Luke Zettlemoyer. 2018. Adversarial example generation with syntactically controlled paraphrase networks. 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 1875–1885, New Orleans, Louisiana. Association for Computational Linguistics.

Unnat Jain, Ziyu Zhang, and Alexander G Schwing. 2017. Creativity: Generating diverse questions using variational autoencoders. In *Computer Vision and Pattern Recognition*.

Melvin Johnson, Mike Schuster, Quoc V Le, Maxim Krikun, Yonghui Wu, Zhifeng Chen, Nikhil Thorat, Fernanda Viégas, Martin Wattenberg, Greg Corrado, et al. 2017. Google's multilingual neural machine translation system: Enabling zero-shot translation. *Transactions of the Association for Computational Linguistics*.

Jad Kabbara and Jackie Chi Kit Cheung. 2016. Stylistic transfer in natural language generation systems using recurrent neural networks. In *Proceedings of the Workshop on Uphill Battles in Language Processing: Scaling Early Achievements to Robust Methods*.

Daniel Kahneman. 2011. *Thinking, fast and slow*. Macmillan.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*.

Divyansh Kaushik, Eduard Hovy, and Zachary C Lipton. 2020. Learning the difference that makes a difference with counterfactually-augmented data. In *Proceedings of the International Conference on Learning Representations*.

Maurice G Kendall. 1938. A new measure of rank correlation. *Biometrika*, 30(1/2):81–93.

Nitish Shirish Keskar, Bryan McCann, Lav R Varshney, Caiming Xiong, and Richard Socher. 2019. Ctrl: A conditional transformer language model for controllable generation. *arXiv preprint arXiv:1909.05858*.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of Empirical Methods in Natural Language Processing*.

Diederik P Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *Proceedings of the International Conference on Learning Representations*.

Guillaume Klein, Yoon Kim, Yuntian Deng, Jean Senellart, and Alexander Rush. 2017. OpenNMT: Open-source toolkit for neural machine translation. In *Proceedings of ACL 2017, System Demonstrations*.

Stanley Kok and Chris Brockett. 2010. Hitting the right paraphrases in good time. In *Conference of the North American Chapter of the Association for Computational Linguistics*. Association for Computational Linguistics.

Ashutosh Kumar, Satwik Bhattamishra, Manik Bhandari, and Partha Talukdar. 2019. Submodular optimization-based diverse paraphrasing and its effectiveness in data augmentation. In *Conference of the North American Chapter of the Association for Computational Linguistics*.

Guillaume Lample, Alexis Conneau, Ludovic Denoyer, and Marc'Aurelio Ranzato. 2018. Unsupervised machine translation using monolingual corpora only. In *Proceedings of the International Conference on Learning Representations*.

J Richard Landis and Gary G Koch. 1977. The measurement of observer agreement for categorical data. *biometrics*, pages 159–174.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*.

Jiwei Li, Will Monroe, and Dan Jurafsky. 2016. A simple, fast diverse decoding algorithm for neural generation. *arXiv preprint arXiv:1611.08562*.

Juncen Li, Robin Jia, He He, and Percy Liang. 2018. Delete, retrieve, generate: a simple approach to sentiment and 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 1865–1874, New Orleans, Louisiana. Association for Computational Linguistics.

Zhuohan Li, Eric Wallace, Sheng Shen, Kevin Lin, Kurt Keutzer, Dan Klein, and Joseph E Gonzalez. 2020. Train large, then compress: Rethinking model size for efficient training and inference of transformers. *arXiv preprint arXiv:2002.11794*.

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 Empirical Methods in Natural Language Processing*.

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. *arXiv preprint arXiv:1907.11692*.

Lajanugen Logeswaran, Honglak Lee, and Samy Bengio. 2018. Content preserving text generation with attribute controls. In *Proceedings of Advances in Neural Information Processing Systems*.

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

Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Effective approaches to attention-based neural machine translation. In *Proceedings of Empirical Methods in Natural Language Processing*.

Yun Ma, Yangbin Chen, Xudong Mao, and Qing Li. 2019. A syntax-aware approach for unsupervised text style transfer. *OpenReview*.

Aman Madaan, Amrith Setlur, Tanmay Parekh, Barnabas Poczos, Graham Neubig, Yiming Yang, Ruslan Salakhutdinov, Alan W Black, and Shrimai Prabhumoye. 2020. Politeness transfer: A tag and generate approach. In *Proceedings of the Association for Computational Linguistics*.

M. Meyerhoff. 2015. *Introducing Sociolinguistics*. Taylor & Francis.

Remi Mir, Bjarke Felbo, Nick Obradovich, and Iyad Rahwan. 2019. Evaluating style transfer for text. In *Conference of the North American Chapter of the Association for Computational Linguistics*.

Shuyo Nakatani. 2010. Language detection library for java.

Jekaterina Novikova, Ondřej Dušek, Amanda Cercas Curry, and Verena Rieser. 2017. Why we need new evaluation metrics for NLG. In *Proceedings of Empirical Methods in Natural Language Processing*.

Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. 2019. fairseq: A fast, extensible toolkit for sequence modeling. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*.

Richard Yuanzhe Pang. 2019. Towards actual (not operational) textual style transfer auto-evaluation. In *Proceedings of the 5th Workshop on Noisy User-generated Text (W-NUT 2019)*.

Richard Yuanzhe Pang and Kevin Gimpel. 2019. Unsupervised evaluation metrics and learning criteria for non-parallel textual transfer. In *Proceedings of the 3rd Workshop on Neural Generation and Translation*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the Association for Computational Linguistics*. Association for Computational Linguistics.

Sunghyun Park, Seung-won Hwang, Fuxiang Chen, Jaegul Choo, Jung-Woo Ha, Sunghun Kim, and Jinyeong Yim. 2019. Paraphrase diversification using counterfactual debiasing. In *Association for the Advancement of Artificial Intelligence*.

Hao Peng, Ankur Parikh, Manaal Faruqui, Bhuwan Dhingra, and Dipanjan Das. 2019. Text generation with exemplar-based adaptive decoding. In *Conference of the North American Chapter of the Association for Computational Linguistics*, Minneapolis, Minnesota. Association for Computational Linguistics.

Shrimai Prabhumoye, Yulia Tsvetkov, Ruslan Salakhutdinov, and Alan W Black. 2018. Style transfer through back-translation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 866–876, Melbourne, Australia. Association for Computational Linguistics.

Daniel Preotiuc-Pietro, Wei Xu, and Lyle Ungar. 2016. Discovering user attribute stylistic differences via paraphrasing. In *Association for the Advancement of Artificial Intelligence*.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI Blog*, 1(8).

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv preprint arXiv:1910.10683*.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of Empirical Methods in Natural Language Processing*.

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 *Conference of the North American Chapter of the Association for Computational Linguistics*.

Cicero Nogueira dos Santos, Igor Melnyk, and Inkit Padhi. 2018. Fighting offensive language on social media with unsupervised text style transfer. In *Proceedings of the Association for Computational Linguistics*.

Thibault Sellam, Dipanjan Das, and Ankur Parikh. 2020. BLEURT: Learning robust metrics for text generation. In *Proceedings of the Association for Computational Linguistics*.

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, and Tommi Jaakkola. 2017. Style transfer from non-parallel text by cross-alignment. In *Advances in neural information processing systems*, pages 6830–6841.

Rakshith Shetty, Bernt Schiele, and Mario Fritz. 2018. A4nt: author attribute anonymity by adversarial training of neural machine translation. In *27th {USENIX} Security Symposium ({USENIX} Security 18)*, pages 1633–1650.

Sandeep Subramanian, Guillaume Lample, Eric Michael Smith, Ludovic Denoyer, Marc'Aurelio Ranzato, and Y-Lan Boureau. 2019. Multiple-attribute text style transfer. In *Proceedings of the International Conference on Learning Representations*.

Akhilesh Sudhakar, Bhargav Upadhyay, and Arjun Maheswaran. 2019. "transforming" delete, retrieve, generate approach for controlled text style transfer. In *Proceedings of Empirical Methods in Natural Language Processing*.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Proceedings of Advances in Neural Information Processing Systems*, pages 3104–3112.

Bakhtiyar Syed, Gaurav Verma, Balaji Vasan Srinivasan, Vasudeva Varma, et al. 2020. Adapting language models for non-parallel author-stylized rewriting. In *Association for the Advancement of Artificial Intelligence*.

Alexey Tikhonov, Viacheslav Shibaev, Aleksander Nagaev, Aigul Nugmanova, and Ivan P Yamshchikov. 2019. Style transfer for texts: Retrain, report errors, compare with rewrites. In *Proceedings of Empirical Methods in Natural Language Processing*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proceedings of Advances in Neural Information Processing Systems*, pages 5998–6008.

Ashwin K Vijayakumar, Michael Cogswell, Ramprasaath R Selvaraju, Qing Sun, Stefan Lee, David Crandall, and Dhruv Batra. 2018. Diverse beam search for improved description of complex scenes. In *Association for the Advancement of Artificial Intelligence*.

Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. 2008. Extracting and composing robust features with denoising autoencoders. In *Proceedings of the International Conference of Machine Learning*, pages 1096–1103.

Alex Wang, Kyunghyun Cho, and Mike Lewis. 2020. Asking and answering questions to evaluate the factual consistency of summaries. In *Proceedings of the Association for Computational Linguistics*.

Alex Warstadt, Amanpreet Singh, and Samuel R Bowman. 2019. Neural network acceptability judgments. *Transactions of the Association for Computational Linguistics*.

John Wieting, Taylor Berg-Kirkpatrick, Kevin Gimpel, and Graham Neubig. 2019. Beyond BLEU:training neural machine translation with semantic similarity. In *Proceedings of the Association for Computational Linguistics*.

John Wieting and Kevin Gimpel. 2018. ParaNMT-50M: Pushing the limits of paraphrastic sentence embeddings with millions of machine translations. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 451–462, Melbourne, Australia. Association for Computational Linguistics.

Sam Wiseman, Stuart Shieber, and Alexander Rush. 2018. Learning neural templates for text generation. In *Proceedings of Empirical Methods in Natural Language Processing*.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, R'emi Louf, Morgan Funtowicz, and Jamie Brew. 2019. HuggingFace's Transformers: State-of-the-art natural language processing. *ArXiv*, abs/1910.03771.

Thomas Wolf, Victor Sanh, Julien Chaumond, and Clement Delangue. 2018. Transfertransfo: A transfer learning approach for neural network based conversational agents. In *NeurIPS CAI Workshop*.

Qizhe Xie, Zihang Dai, Eduard Hovy, Minh-Thang Luong, and Quoc V Le. 2019. Unsupervised data augmentation for consistency training. *arXiv preprint arXiv:1904.12848*.

Jingjing Xu, Xu Sun, Qi Zeng, Xiaodong Zhang, Xuancheng Ren, Houfeng Wang, and Wenjie Li. 2018. Unpaired sentiment-to-sentiment translation: A cycled reinforcement learning approach. In *Proceedings of the Association for Computational Linguistics*.

Wei Xu, Chris Callison-Burch, and Courtney Napoles. 2015. Problems in current text simplification research: New data can help. *Transactions of the Association for Computational Linguistics*, 3:283–297.

Wei Xu, Alan Ritter, Bill Dolan, Ralph Grishman, and Colin Cherry. 2012. Paraphrasing for style. In *Proceedings of International Conference on Computational Linguistics*.

Zichao Yang, Zhiting Hu, Chris Dyer, Eric P Xing, and Taylor Berg-Kirkpatrick. 2018. Unsupervised text style transfer using language models as discriminators. In *Proceedings of Advances in Neural Information Processing Systems*.

Kuo-Hao Zeng, Mohammad Shoeybi, and Ming-Yu Liu. 2020. Style example-guided text generation using generative adversarial transformers. *arXiv preprint arXiv:2003.00674*.

Zhirui Zhang, Shuo Ren, Shujie Liu, Jianyong Wang, Peng Chen, Mu Li, Ming Zhou, and Enhong Chen. 2018. Style transfer as unsupervised machine translation. *arXiv preprint arXiv:1808.07894*.

Daniel M Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B Brown, Alec Radford, Dario Amodei, Paul Christiano, and Geoffrey Irving. 2019. Fine-tuning language models from human preferences. *arXiv preprint arXiv:1909.08593*.

A Appendices for "Reformulating Unsupervised Style Transfer as Paraphrase Generation"

A.1 PARANMT-50M Filtering Details

We train our paraphrase model in a seq2seq fashion using the PARANMT-50M corpus (Wieting and Gimpel, 2018), which was constructed by back-translating (Sennrich et al., 2016) the Czech side of the CzEng parallel corpus (Bojar et al., 2016). This corpus is large and noisy and we aggressively filter it to encourage content preservation and diversity maximization. We use the following filtering,

Content Filtering: We remove all sentence pairs which score lower than 0.5 on a strong paraphrase similarity model from Wieting et al. (2019).[20] We filter sentence pairs by length, allowing a maximum length difference of 5 words between paired sentences. Finally, we remove very short and long sentences by only keeping sentence pairs with an average token length between 7 and 25.

Lexical Diversity Filtering: We only preserve backtranslated pairs with sufficient unigram distribution difference. We filter all pairs where more than 50% of words in the backtranslated sentence can be found in the source sentence. This is computed using the SQuAD evaluation scripts (Rajpurkar et al., 2016). Additionally, we remove sentences with more than 70% trigram overlap.

Syntactic Diversity Filtering: We discard all paraphrases which have a similar word ordering. We compare the relative ordering of the words shared between the input and backtranslated sentence by measuring the Kendall tau distance (Kendall, 1938) or the "bubble-sort" distance. We keep all backtranslated pairs which are at least 50% shuffled.[21]

LangID Filtering: Finally, we discard all sentences where both the input and backtranslated sentence are classified as non-English using `langdetect`.[22]

Effect of each filter: We adopt a pipelined approach to filtering. The PARANMT-50M corpus size after each stage of filtering is shown in Table 8.

	Filter Stage	Corpus Size
0.	Original	51.41M
1.	Content Similarity	30.49M
2.	Trigram Diversity	9.03M
3.	Unigram Diversity	1.96M
4.	Kendall-Tau Diversity	112.01K
5.	Length Difference	82.64K
6.	LangID	74.55K

Table 8: Steps of filtering conducted on PARANMT-50M along with its effect on corpus size.

A.2 Generative Model Details

This section provides details of our seq2seq model used for both paraphrase model and style-specific inverse paraphrase model. Recent work (Radford et al., 2019) has shown that GPT2, a massive transformer trained on a large corpus of unlabeled text using the language modeling objective, is very effective in performing more human-like text generation. We leverage the publicly available GPT2-large checkpoints by finetuning it on our custom datasets with a small learning rate. However, GPT2 is an unconditional language model having only a decoder network, and traditional seq2seq setups use separate encoder and decoder neural network (Sutskever et al., 2014) with attention (Bahdanau et al., 2014). To avoid training an encoder network from scratch, we use the encoder-free seq2seq modeling approach described in Wolf et al. (2018). where both input and output sequences are fed to the decoder network separated with a special token, and use separate segment embeddings. Our model is implemented using the `transformers` library[23] (Wolf et al., 2019). We use encoder-free seq2seq modeling (Wolf et al., 2018) which feeds the input into the decoder neural network, separating it with segment embeddings. We fine-tune GPT2-large to perform encoder-free seq2seq modeling.

Architecture: Let $\mathbf{x} = (x_1, ..., x_n)$ represent the tokens in the input sequence and let $\mathbf{y} = (y_{bos}, y_1, ..., y_m, y_{eos})$ represent the tokens of the output sequence, where y_{bos} and y_{eos} corresponds to special beginning and end of sentence tokens. We feed the sequence $(x_1, ..., x_n, y_{bos}, y_1, ..., y_m)$ as input to GPT2 and train it on the next-word prediction objective for the tokens $y_1, ..., y_m, y_{eos}$

[20] We use the SIM model from Wieting et al. (2019), which achieves a strong performance on the SemEval semantic text similarity (STS) benchmarks (Agirre et al., 2016)

[21] An identical ordering of words is 0% shuffled whereas a reverse ordering is 100% shuffled.

[22] This is using the Python port of Nakatani (2010), https://github.com/Mimino666/langdetect.

[23] https://github.com/huggingface/transformers

using the cross-entropy loss. During inference, the sequence $(x_1, ..., x_n, y_{bos})$ is fed as input and the tokens are generated in an autoregressive manner (Vaswani et al., 2017) until y_{eos} is generated.

Every token in $\mathbf{x}$ and $\mathbf{y}$ is passed through a shared input embedding layer to obtain a vector representation of every token. To encode positional and segment information, learnable positional and segment embeddings are added to the input embedding consistent with the GPT2 architecture. Segment embeddings are used to denote whether a token belongs to sequence $\mathbf{x}$ or $\mathbf{y}$.

Other seq2seq alternatives: Note that our unsupervised style transfer algorithm is agnostic to the specific choice of seq2seq modeling. We wanted to perform transfer learning from massive left-to-right language models like GPT2, and found the encoder-free seq2seq approach simple and effective. Future work includes finetuning more recent models like T5 (Raffel et al., 2019) or BART (Lewis et al., 2019). These models use the standard seq2seq setup of separate encoder / decoder networks and pretrain them jointly using denoising autoencoding objectives based on language modeling.

Hyperparameter Details: We finetune GPT2-large using NVIDIA TESLA M40 GPUs for 2 epochs using early stopping based on validation set perplexity. The models are finetuned using a small learning rate of 5e-5 and converge to a good solution fairly quickly as noticed by recent work (Li et al., 2020; Kaplan et al., 2020). Specifically, each experiment completed within a day of training on a single GPU, and many experiments with small datasets took a lot less time. We use a minibatch size of 10 sentence pairs and truncate sequences which are longer than 50 subwords in the input or output space. We use the Adam optimizer (Kingma and Ba, 2015) with the weight decay fix and using a linear learning rate decay schedule, as implemented in the `transformers` library. Finally, we left-pad the input sequence to get a total input length of 50 subwords and right-pad output sequence to get a total output length of 50 subwords. This special batching is necessary to use minibatches during inference time. Special symbols are used to pad the sequences and they are not considered in the cross-entropy loss. Our model has 774M trainable parameters, identical to the original GPT2-large.

A.3 Classifier Model Details

We fine-tune RoBERTa-large to build our classifier, using the official implementation in `fairseq`. We use a learning rate of 1e-5 for all experiments with a minibatch size of 32. All models were trained on a single NVIDIA RTX 2080ti GPU, with gradient accumulation to allow larger batch sizes. We train models for 10 epochs and use early stopping on the validation split accuracy. We use the Adam optimizer (Kingma and Ba, 2015) with modifications suggested in the RoBERTa paper (Liu et al., 2019). Consistent with the suggested hyperparameters, we use a learning rate warm-up for the first 6% of the updates and then decay the learning rate.

A.4 OpenNMT Model Details

We train sequence-to-sequence models with attention based on LSTMs using OpenNMT (Klein et al., 2017) using their PyTorch port.[24] We mostly used the default hyperparameter settings of `OpenNMT-py`. The only hyperparameter we modified was the learning rate schedule, since our datasets were small and overfit quickly. For the paraphrase model, we started decay after 11000 steps and halved the learning rate every 1000 steps. For Shakespeare, we started the decay after 3000 steps and halved the learning rate every 500 steps. For Formality, we started the decay after 6000 steps and halved the learning rate every 1000 steps. These modifications only slightly improved validation perplexity (by 3-4 points in each case).

We used early stopping on validation perplexity and checkpoint the model every 500 optimization steps. The other hyperparameters are the default `OpenNMT-py` settings — SGD optimization using learning rate 1.0, LSTM seq2seq model with global attention (Luong et al., 2015), 500 hidden units and embedding dimensions and 2 layers each in the encoder and decoder.

A.5 More Comparisons with Prior Work

Please refer to Table 12 for an equivalent of Table 1 using BLEU scores.

We present more comparisons with prior work in Table 13. We use the generated outputs for the Formality test set available in the public repository of Luo et al. (2019) (including outputs from the algorithms described in Prabhumoye et al., 2018 and Li et al., 2018) and run them on our evaluation pipeline. We compare the results

[24] https://github.com/OpenNMT/OpenNMT-py

with our formality transfer model used in Table 1 and Table 2. We note significant performance improvements, especially in the fluency of the generated text. Note that there is a domain shift for our model, since we trained our model using the splits of He et al. (2020) which use the Entertainment & Music splits of the Formality corpus. The outputs in the repository of Luo et al. (2019) use the Family & Relationships split. It is unclear in the paper of Luo et al. (2019) whether the models were trained on the Family & Relationships training split or not.

Other Comparisons: We tried to compare against other recent work in style transfer based on Transformers, such as Dai et al. (2019) and Sudhakar et al. (2019). Both papers do not evaluate their models on the datasets we use (Shakespeare and Formality), where parallel sentences preserve semantics.

The only datasets used in Dai et al. (2019) were sentiment transfer benchmarks, which modify semantic properties of the sentence. We attempted to train the models in Dai et al. (2019) using their codebase on the Shakespeare dataset, but faced three major issues 1) missing number of epochs / iterations. The early stopping criteria is not implemented or specified, and metrics were being computed on the *test set* every 25 training iterations, which is invalid practice for choosing the optimal checkpoint; 2) specificity of the codebase to the Yelp sentiment transfer dataset in terms of maximum sequence length and evaluation, making it non-trivial to use for any other dataset; 3) despite our best efforts we could not get the model to converge to a good minima which would produce fluent text (besides word-by-word copying) when trained on the Shakespeare dataset.

Similarly, the datasets used in Sudhakar et al. (2019) modify semantic properties (sentiment, political slant etc.). On running their codebase on the Shakespeare dataset using the default hyperparameters, we achieved a poor performance of 53.1% ACC, 55.2 SIM and 56.5% FL, aggregating to a $J(\text{A,S,F})$ score of 18.4. Similarly on the Formality dataset, performance was poor with 41.7% ACC, 67.8 SIM and 67.7% FL, aggregating to $J(\text{A,S,F})$ score of 18.1. A qualitatively inspection showed very little abstraction and nearly word-by-word copying from the input (due to the delete & generate nature of the approach), which explains the higher SIM score but lower ACC score (just like COPY baseline in Table 1). Fluency was low despite GPT pretraining, perhaps due to the token deletion step in the algorithm.

A.6 Details of our Dataset, CDS

We provide details of our sources, the sizes of individual style corpora and examples from our new benchmark dataset CDS in Table 14. We individually preprocessed each corpus to remove very short and long sentences, boilerplate text (common in Project Gutenberg articles) and section headings. We have added some representative examples from each style in Table 14. More representative examples (along with our entire dataset) will be provided in the project page `http://style.cs.umass.edu`.

Style Similarity: In Figure 4 we plot the cosine similarity between styles using the averaged `[CLS]` vector of the trained RoBERTa-large classifier (inference over validation set). The off-diagonal elements show intuitive domain similarities, such as (Lyrics, Poetry); (AAE, Tweets); (Joyce, Shakespeare) or among classes from the Corpus of Historical American English.

A.7 Diverse Paraphrasing on CDS

We compare the quality and diversity of the paraphrases generated by our diverse and non-diverse paraphrasers on our dataset CDS in Table 16. Note that this is the pseudo parallel training data for the inverse paraphrase model (described in Section 2.1 and Section 2.4) and not the actual style transferred sentences. Overall, the diverse paraphraser achieves high diversity, with 51% unigram change and 27% word shuffling,[25] compared to 28% unigram and 6% shuffling for non-diverse paraphraser, while maintaining good semantic similarity (SIM= 72.5 vs 83.9 for non-diverse) even in complex stylistic settings.

A.8 Style Transfer Performance on CDS

We provide a detailed breakdown of performance in different styles of CDS in Table 15. For each of the 11 target styles, we style transferred 1,000 sentences from every other style and jointly evaluated the 10,000 generations. Some styles are more successfully transferred than others, such as Switchboard, Lyrics and James Joyce. While wearing the

[25] The "unigram change" and "word shuffling" refer to the unigram F1 word overlap and Kendall's τ_B scores.

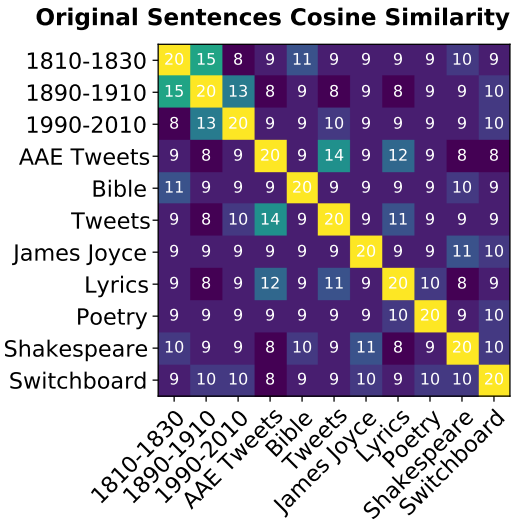

Figure 4: Cosine similarities between styles in CDS using the [CLS] vectors of the RoBERTa-large classifier (normalized to $[0, 20]$). The off-diagonal elements show intuitive domain similarities, such as (Lyrics, Poetry); (AAE, Tweets); (Joyce, Shakespeare) or among classes from the COHA corpus.

p value for nucleus sampling, we notice a trend similar to the **Nucleus sampling trades off ACC for SIM** experiment in Section 5. Increasing the p value improves ACC at the cost of SIM. However unlike the Shakespeare and Formality dataset, we find $p = 0.6$ the optimal value for the best ACC-SIM tradeoff.

Note that Fluency scores on this dataset could be misleading since even the original sentences from some styles are often classified as disfluent (Orig. FL). Qualitatively, this seems to happen for styles with rich lexical and syntactic diversity (like Romantic Poetry, James Joyce). These styles tend to be out-of-distribution for the fluency classifier trained on the CoLA dataset (Warstadt et al., 2019).

A.9 A Survey of Evaluation Methods

We present a detailed breakdown of evaluation metrics used in prior work in Table 10 and the implementations of the metrics in Table 11. Notably, only 3 out of 23 prior works use an absolute sentence-level aggregation evaluation. Other works either perform "overall A/B" testing, flawed corpus-level aggregation or don't perform any aggregation at all. Note that while "overall A/B" testing cannot be gamed like corpus-aggregation, it has a few issues — (1) it is a *relative* evaluation and does not provided an *absolute* performance score for future reference; (2) "A/B" testing requires human evaluation, which is expensive and noisy; (3) evaluating overall performance will require human annotators to be familiar with the styles and style transfer task setup; (4) Kahneman (2011) has shown that asking humans to give a single number for "overall score" is biased when compared to an aggregation of *independent* scores on different metrics. Luckily, the sentence-level aggregation in Li et al. (2018) does the latter and is the closest equivalent to our proposed $J(\cdot)$ metric.

A.10 Details on Human Evaluation

We conduct experiments of Amazon Mechanical Turk, annotating the paraphrase similarity of 150 sentences with 3 annotators each. We report the label chosen by two or more annotators, and collect additional annotations in the case of total disagreement. We pay workers 5 cents per sentence pair ($10-15 / hr). We only hire workers from USA, UK and Australia with a 95% or higher approval rating and at least 1000 approved HITs. Sentences where the input was exactly copied (after lower-casing and removing punctuation) are automatically assigned the option **2** paraphrase and grammatical. Even though these sentences are clearly not style transferred, we expect them to be penalized in J(ACC,SIM,FL) by poor ACC. We found that every experiment had a Fleiss kappa (Fleiss, 1971) of at least 0.13 and up to 0.45 (slight to moderate agreement according to (Landis and Koch, 1977)). A qualitative inspection showed that crowdworkers found it easier to judge sentence pairs in the Formality dataset than Shakespeare, presumably due to greater familiarity with modern English. We also note that crowdworkers had higher agreement for sentences which were clearly not paraphrases (like the UNMT / DLSM generations on the Formality dataset).

Calculating Metrics in Table 2: To calculate SIM, we count the percentage of sentences which humans assigned a label **1** (ungrammatical paraphrase) or **2** (grammatical paraphrase). This is used as a binary value to calculate J(ACC, SIM). To calculate J(ACC, SIM, FL), we count sentences which are correctly classified as well as humans assigned a label of **2** (grammatical paraphrase). We cannot calculate FL alone using the popular 3-way evaluation, since the fluent sentences which are not paraphrases are not recorded.

A.11 More Example Generations

More examples are provided in Table 9. All of our style transferred outputs on CDS will be available in the project page of this work, http://style.cs.umass.edu.

A.12 More Related Work

Our inverse paraphrase model is a style-**controlled text generator** which automatically learns lexical and syntactic properties prevalent in the style's corpus. Explicit syntactically-controlled text generation has been studied previously using labels such as constituency parse templates (Iyyer et al., 2018; Akoury et al., 2019) or learned discrete latent templates (Wiseman et al., 2018). Syntax can also be controlled using an exemplar sentence (Chen et al., 2019; Guu et al., 2018; Peng et al., 2019). While style transfer requires the underlying content to be provided as input, another direction explores attribute-controlled *unconditional* text generation (Dathathri et al., 2020; Keskar et al., 2019; Zeng et al., 2020; Ziegler et al., 2019).

Diversity in text generation is often encouraged during inference time via heuristic modifications to beam search (Li et al., 2016; Vijayakumar et al., 2018), nucleus sampling (Holtzman et al., 2020) or submodular optimization (Kumar et al., 2019); in contrast, we simply filter our training data to increase diversity. Other algorithms learn to condition generation on latent variables during training (Bowman et al., 2016), which are sampled from at inference time to encourage diversity (Jain et al., 2017; Gupta et al., 2018; Park et al., 2019). Relatedly, Goyal and Durrett (2020) promote syntactic diversity of paraphrases by conditioning over possible syntactic rearrangements of the input.

Direction	Input	Output
1810. → Swit.	She forgot her gold pieces lying on the table in front of her.	and uh she forgot the gold pieces on the desk
1810. → AAE	This story is not well devised, my lord.	Lord this story ain't no good
Poetry → Swit.	Why should our young Endymion pine away!	i don't know why we should have such a letdown for our young Endymion
Tweet → Shak.	This is an insult to the Munster's.	Tis an affront to the Munster.
Swit. → 1810.	and my lawn is very green and my flowers are still blooming	I have plenty of green turf and flowers here.
1890. → AAE	I want no pursuit when I do my work.	I dont wanna chase when im at work
1990. → Joyce	"You're scaring me," Clara says .	'Tis scare me, says Clara.
Tweet → Bible	Can we build AI without losing control over it?	Can a man build a computer without losing his power?
AAE → 1810.	I Use To Think They Were Ugly.	I thought them odious–and hateful.
Bible → 1990.	hath thy soul lothed Zion?	Is Zion a place where you have your soul?
Bible → Tweet	This persuasion cometh not of him that calleth you.	Its not him who comes from. Its who you call
Swit. → AAE	um yeah just a second	yea just a lil bit
Joyce → Shak.	Smell of burn.	This is the smoke of a burning.
1990. → Joyce	The Second World War, son, " he answered.	SECOND WORLD WAR, son, says he.
1990. → AAE	no, I implore you not to consider this option.	Naw I asked u not to think bout that option
Tweet → Joyce	i wanted to sleep but now i cant	sleep is my preferred repose.
1990. → 1810.	She ninety-three and she can't hear too good.	She is 93 years of age and cannot hear well yet.
AAE → Poetry	y'all play? we spray	And weel weel, we'll spray a' that.
Joyce → Poetry	Now, don't tell a big fib!	And say not so big a lie!
1810. → Bible	without first veiling that, I could not sin.	And I could not sin without veil.
1890. → Shak.	When did we ever find Gaston de Marignac objectionable?	When was 't that we found Gaston de Marignac's foul faults?
Shak. → 1990.	O, much I fear some ill unlucky thing.	I was afraid of a lot of misadventures.
Shak. → 1810.	By Jesu, a very good blade!	The very knife of heaven, Jesu!
Lyrics → 1890	For a thing to explode	Oh, oh, oh, for something to blow up!
Lyrics → Bible	You are the sun that lights my world	Thou art the sun that lighteth mine earth: thou art my righteousness.
Lyrics → Joyce	But now it shouldn't have to be this way	But oughtn't to be like that.
Swit. → 1810.	oh that poor child	(sighing) the poor dear little thing.
Swit. → Lyrics	you know Karen i wonder if we are recording	Karen, I wonder if we're tapein'
Tweet → Bible	Ok will get back to you ASAP!	And he said, All right; I will return.
Tweet → Shak.	Movies are always funnier in the cinema.	It always is more humorous in the theatre.
Lyrics → Poetry	You know I still fear the bomb	I was yet a-fear'd at the bomb,
Joyce → 1890.	Just missed that by a hair.	It was a hair-spitting thing I missed.
Joyce → 1810.	Me seemeth a dragon man.	Hark ye, I'm a dragon man, Me.
Poetry → Bible	Of deities or mortals, or of both,	Of the gods, and of the people, or of the both of them together.
Poetry → Lyrics	Makes countless thousands mourn!	Gonna be a lot of people mourning!
Poetry → Bible	My helpless lambs, I trust them wi' him.	Surely he will feed my lambs in safety.
1890. → Poetry	Stark was sentenced to ten years' imprisonment, Gibbon to five.	Stark was sentence'd ten yeares, Gibbon five;
1890. → Swit.	I want no pursuit when I do my work.	i don't want to chase after when i'm at work
Tweet → Bible	The way this is set up has me dying	This death cometh because of this.
Tweet → Swit.	Yay! Ill order them then! I dont have patience to wait on them!	now i'm gonna order those
Lyrics → Joyce	And all the guns is in Kesha's name	All your gunsees belong to Kesha.
AAE → 1810.	boredom gonna be the death of me one day	I shall die, one day, of the insupportable want of amusement.
AAE → 1890.	That's just what I needed to see.... Thank Ya Lord	Thank you, Lord; that is just what I was expecting.
AAE → Swit.	okay ii will see you later	yeah see you later bye
Poetry → Tweet	Fam'd heroes! had their royal home:	royal bloods heroes:
Tweet → Bible	Check out this new painting that I uploaded to!	Look upon my new picture that I have set before thee!
Swit. → Shak.	so uh what do you wear to work	And what dost thou wear for thy work?
Tweet → Poetry	Now I gotta delete it	O now, must I part? And can I now erase
Tweet → 1810.	#India is now producing the worlds cheapest solar power #energy	Now is India's solar power cheapest of all the world.
Poetry → Joyce	Away, away, or I shall dearly rue	O offside, away, or do I am rather sad.
Tweet → Swit.	Oh shit ima be a senior	so uh i got to the senior level of the business

Table 9: More example outputs from our model STRAP trained on our dataset CDS. Our project page will provide all 110k style transferred outputs generated by STRAP on CDS.

Paper	Automatic					Human				
	ACC	SIM	FL	CA	SA	ACC	SIM	FL	CA	SA
Hu et al. (2017)	✓									
Shen et al. (2017)	✓					✓		✓		A/B
Shetty et al. (2018)	✓						A/B			
Fu et al. (2018)	✓	✓					✓			
Li et al. (2018)	✓	✓				✓	✓	✓		✓
Zhang et al. (2018)	✓	✓				✓	✓	✓		✓
Nogueira dos Santos et al. (2018)	✓	✓	✓							
Prabhumoye et al. (2018)	✓						A/B	✓		
Xu et al. (2018)	✓	✓		✓		✓	✓		✓	
Logeswaran et al. (2018)	✓	✓	✓			✓	✓	✓		
Yang et al. (2018)	✓	✓	✓							
Subramanian et al. (2019)	✓	✓	✓			✓	✓	✓		A/B
Luo et al. (2019)	✓	✓		✓		✓	✓	✓	✓	✓
Pang and Gimpel (2019)	✓	✓	✓	✓		A/B	A/B	A/B		A/B
Ma et al. (2019)	✓	✓	✓			✓	✓	✓		
Dai et al. (2019)	✓	✓	✓			A/B	A/B	A/B		
Sudhakar et al. (2019)	✓	✓	✓			A/B	A/B	A/B		A/B
Mir et al. (2019)	✓	✓	✓			✓	✓	✓		
Gröndahl and Asokan (2019)	✓	✓					✓			
Tikhonov et al. (2019)	✓	✓								
Syed et al. (2020)	✓	✓								
Madaan et al. (2020)	✓	✓				✓	✓	✓		
He et al. (2020)	✓	✓	✓							
Ours	✓	✓	✓	✓	✓		✓	✓		✓

Table 10: Survey of evaluation methods used in 23 prior papers. We check whether prior work evaluate their algorithm on transfer accuracy (ACC), semantic similarity (SIM), fluency (FL), corpus-level aggregation (CA) and sentence-level aggregation (SA). We use the "A/B" to denote relative comparisons via A/B testing between generations from the baseline and the proposed system, rather than absolute performance numbers. Specific implementations of the metrics have been provided in Table 11. We do not include Pang (2019) since it's a survey of existing evaluation methods.

Paper	Automatic			Human		
	ACC	SIM	FL	ACC	SIM	FL
Hu et al. (2017)	L-CNN					
Shen et al. (2017)	CNN			Likert-4		Likert-4
Shetty et al. (2018)	RNN/CNN	METEOR			A/B	
Fu et al. (2018)	LSTM	GloVE			Likert-3	
Li et al. (2018)	LSTM	BLEU		Likert-5	Likert-5	Likert-5
Zhang et al. (2018)	GRU	BLEU		Likert-5	Likert-5	Likert-5
Nogueira dos Santos et al. (2018)	SVM	GloVE	PPL			
Prabhumoye et al. (2018)	CNN				A/B	Likert-4
Xu et al. (2018)	CNN	BLEU		Likert-10	Likert-10	
Logeswaran et al. (2018)	CNN	BLEU	PPL	Likert-5	Likert-5	Likert-5
Yang et al. (2018)	CNN	BLEU	PPL			
Subramanian et al. (2019)	fastText	BLEU	PPL	Binary	Likert-5	Likert-5
Luo et al. (2019)	CNN	BLEU		Likert-5	Likert-5	Likert-5
Pang and Gimpel (2019)	CNN	GloVE	PPL	A/B	A/B	A/B
Ma et al. (2019)	CNN	BLEU	PPL	Likert-5	Likert-5	Likert-5
Dai et al. (2019)	fastText	BLEU	PPL	A/B	A/B	A/B
Sudhakar et al. (2019)	fastText	GLEU	PPL	A/B	A/B	A/B
Mir et al. (2019)	EMD	GloVE*	Classify	Likert-5*	Likert-5*	Binary*
Gröndahl and Asokan (2019)	LSTM/CNN	METEOR				
Tikhonov et al. (2019)	CNN	BLEU				
Syed et al. (2020)	FineGrain	BLEU				
Madaan et al. (2020)	AWD-LSTM	METEOR		Likert-5	Likert-5	Likert-5
He et al. (2020)	CNN	BLEU	PPL			
Ours	RoBERTa-L	SIM-PP	Classify		Binary	Binary

Table 11: Survey of implementations of evaluation metrics to measure Accuracy (ACC), Similarity (SIM) and Fluency (FL) used in 23 prior papers. For a cleaner version of this table with aggregation information, see Table 10. The * marks in Mir et al. (2019) denote a carefully designed unique implementation. We do not include Pang (2019) since it's a survey of existing evaluation methods.

Model	Formality					Shakespeare				
	ACC	SIM	FL	GM(A,S,F)	J(A,S,F)	ACC	SIM	FL	GM(A,S,F)	J(A,S,F)
COPY	5.2	41.8	88.4	26.8	0.2	9.6	20.1	79.1	24.8	0.1
NAÏVE	49.7	22.1	89.4	44.4	2.4	49.9	10.5	78.9	34.6	1.1
REF	93.3	100	89.7	94.2	88.2	90.4	100	79.1	89.4	67.2
UNMT	78.5	15.1	52.5	39.7	11.7	70.5	7.9	49.6	30.2	1.7
DLSM	78.0	18.5	53.7	42.6	9.5	71.1	12.5	49.4	35.2	2.0
STRAP ($p = 0.0$)	67.7	28.8	90.4	56.1	19.3	71.7	10.3	85.2	39.8	5.9
STRAP ($p = 0.6$)	70.7	25.3	88.5	54.1	17.2	75.7	8.8	82.7	38.1	5.4
STRAP ($p = 0.9$)	76.8	17.0	77.4	46.6	12.2	79.8	6.1	71.7	32.7	3.4

Table 12: A table equivalent to Table 1 but using BLEU scores for SIM instead of the paraphrase similarity model from Wieting et al. (2019). The Formality dataset had 4 available reference sentences whereas the Shakespeare dataset had only 1 available reference sentence. Our system STRAP significantly beats prior work (UNMT, DLSM) on $J(\cdot)$ metrics even with BLEU scores.

Model	ACC (A)	SIM (S)		FL (F)	J(A,S)		J(A,S,F)	
		BL	PP		BL	PP	BL	PP
COPY	8.0	32.6	80.9	90.1	0.4	7.1	0.3	6.4
REF	87.8	100	100	90.1	91.1	87.8	83.5	78.9
NAÏVE	67.9	10.7	32.0	91.5	1.7	9.3	1.5	8.5
BT (Prabhumoye et al., 2018)	47.4	1.3	21.1	8.0	0.7	11.4	0.0	1.3
MultiDec (Fu et al., 2018)	26.0	12.0	36.9	15.1	1.4	8.9	0.0	1.5
Del. (Li et al., 2018)	24.2	30.1	53.5	20.8	3.1	10.2	0.0	1.6
Unpaired (Xu et al., 2018)	53.9	1.6	16.3	34.9	0.4	10.9	0.0	2.2
DelRetri. (Li et al., 2018)	52.8	21.9	47.6	16.3	11.9	23.4	0.2	4.2
CrossAlign. (Shen et al., 2017)	59.0	3.3	25.0	31.7	2.0	14.9	0.3	5.2
Retri. (Li et al., 2018)	90.0	0.5	9.0	62.1	0.5	8.3	0.3	5.5
Templ. (Li et al., 2018)	37.1	36.4	67.8	32.3	11.9	23.7	1.3	7.8
DualRL (Luo et al., 2019)	51.8	45.0	65.1	59.0	14.6	29.9	8.1	21.7
UNMT (Zhang et al., 2018)	64.5	34.4	64.8	45.9	28.2	41.2	14.7	22.1
STRAP ($p = 0.0$)*	57.7	31.1	69.7	93.8	19.5	40.8	**18.3**	38.7
STRAP ($p = 0.6$)*	63.4	26.5	66.7	91.4	18.3	**43.0**	17.1	**40.0**
STRAP ($p = 0.9$)*	70.3	17.3	59.0	81.4	13.6	41.6	11.8	34.3

Table 13: More comparisons against prior work on the Formality dataset (Rao and Tetreault, 2018) using the outputs provided in the publicly available codebase of Luo et al. (2019) using both BLEU score (BL) and paraphrase similarity (PP). This model uses the Family & Relationships split of the Formality dataset whereas (He et al., 2020) used the Entertainment & Music split. Hence, we have retrained our RoBERTa-large classifiers to reflect the new distribution. ***Note**: While our system significantly outperforms prior work, we re-use the formality system used in Table 1 and Table 2 for these results, which was trained on Entertainment & Music (consistent with He et al. (2020)). There could be a training dataset mismatch between our model and the models from Luo et al. (2019), since the Formality dataset has two domains. This is not clarified in Luo et al. (2019) to the best of our knowledge.

Style	Train	Dev	Test	Source	Examples
Shakespeare	24,852	1,313	1,293	Shakespeare split of Xu et al. (2012).	1. *Why, Romeo, art thou mad?* 2. *I beseech you, follow straight.*
English Tweets	5,164,874	39,662	39,690	A random sample of English tweets collected on 8th-9th July, 2019 using Twitter APIs.	1. *Lol figures why I dont wanna talk to anyone rn* 2. *omg no problem i felt bad holding it! i love youuuu*
Bible	31,404	1,714	1,714	The English Bible collected from Project Gutenberg (Hart, 1992) (link).	1. *Jesus saith unto her, Woman, what have I to do with thee?* 2. *Wherefore it is lawful to do well on the sabbath days.*
Romantic Poetry	26,880	1,464	1,470	The Romantic section of the Poetry bookshelf on Project Gutenberg (link).	1. *There in that forest did his great love cease;* 2. *But, oh! for Hogarth's magic pow'r!*
Switchboard	145,823	1,487	1,488	Conversational speech transcripts (link) from the Switchboard speech recognition corpus (Godfrey et al., 1992).	1. *uh-huh well we're not all like that um* 2. *well yes i i well i- i don't think i have the time to really become a student in every article*
AAE (African American English) Tweets	717,634	7,316	7,315	Using the geo-located tweet corpus collected by Blodgett et al. (2016).	1. *ay yall everything good we did dat...* 2. *I know data right, it don't get more real than that.*
James Joyce	37,082	2,054	2,043	Two novels (Ulysses, Finnegans) of James Joyce from Project Gutenberg (link) and the Internet Archive (link).	1. *At last she spotted a weeny weeshy one miles away.* 2. *chees of all chades at the same time as he wags an antomine art of being rude like the boor.*
Lyrics	4,588,522	252,368	252,397	Music lyrics dataset from MetroLyrics, used in a Kaggle competition (link).	1. *I gotta get my mind off you,* 2. *This is it, we are, baby, we are one of a kind*
1810-1830 historical English	205,286	5,340	5,338	1810-1830 in the Corpus of Historical American English (Davies, 2012) using fiction, non-fiction and magazine domains (link).	1. *The fulness of my fancy renders my eye vacant and inactive.* 2. *What then do you come hither for at such an hour?*
1890-1910 historical English	1,210,687	32,024	32,018	1890-1910 in the Corpus of Historical American English using fiction, non-fiction and magazine domains (link).	1. *Nor shall I reveal the name of my friend; I do not wish to expose him to a torrent of abuse.* 2. *You know olive oil don't give the brightest illumination.*
1990-2010 historical English	1,865,687	48,985	48,982	1990-2010 in the Corpus of Historical American English using fiction, non-fiction and magazine domains (link).	1. *They were, in fact, tears of genuine relief.* 2. *I don't know why, but I sensed there was something wrong.*
Total	14,018,731	393,727	393,748		

Table 14: Details of our new benchmark dataset CDS along with representative examples. Our dataset contains eleven lexically and syntactically diverse styles and has a total of nearly 15M sentences, an order of magnitude larger than previous datasets. We will provide more representative examples along with our entire dataset in the project page `http://style.cs.umass.edu`.

Split	Orig. ACC	Orig. FL	Model	ACC (A)	SIM (S)	FL (F)	J(A,S)	J(A,S,F)
AAE Tweets	87.6	56.4	Ours ($p = 0.0$)	21.0	70.1	71.6	12.6	8.3
			Ours ($p = 0.6$)	32.5	65.7	63.5	18.3	**10.2**
			Ours ($p = 0.9$)	46.1	57.8	45.9	**23.6**	9.8
Bible	98.3	87.5	Ours ($p = 0.0$)	48.0	58.4	81.2	24.7	20.9
			Ours ($p = 0.6$)	52.5	55.1	79.8	**25.7**	**21.3**
			Ours ($p = 0.9$)	56.9	49.4	74.0	25.3	19.3
COHA 1810s-1820s	83.0	89.1	Ours ($p = 0.0$)	25.9	66.5	84.5	16.4	13.7
			Ours ($p = 0.6$)	34.0	63.0	81.5	20.1	16.0
			Ours ($p = 0.9$)	42.7	57.3	73.6	**22.9**	**16.5**
COHA 1890s-1900s	76.5	91.2	Ours ($p = 0.0$)	36.1	68.9	86.7	23.7	21.2
			Ours ($p = 0.6$)	41.1	65.7	83.8	**25.5**	**22.1**
			Ours ($p = 0.9$)	44.3	59.4	72.0	25.0	19.2
COHA 1990s-2000s	86.9	96.8	Ours ($p = 0.0$)	40.4	69.0	87.7	26.6	24.4
			Ours ($p = 0.6$)	46.1	65.6	86.0	**28.9**	**26.3**
			Ours ($p = 0.9$)	46.1	59.4	76.1	26.1	21.7
English Tweets	80.7	79.9	Ours ($p = 0.0$)	20.0	71.0	79.1	13.5	11.0
			Ours ($p = 0.6$)	28.9	67.5	72.2	18.1	**13.7**
			Ours ($p = 0.9$)	40.8	60.0	55.5	**22.7**	13.4
James Joyce	87.1	48.2	Ours ($p = 0.0$)	43.0	69.6	79.8	28.7	22.0
			Ours ($p = 0.6$)	52.2	63.7	62.8	32.0	**29.6**
			Ours ($p = 0.9$)	63.6	54.8	40.5	**33.5**	11.3
Lyrics	88.7	78.9	Ours ($p = 0.0$)	51.9	71.6	79.4	**35.6**	**29.0**
			Ours ($p = 0.6$)	53.4	68.6	71.4	34.8	26.0
			Ours ($p = 0.9$)	53.3	62.1	51.9	31.4	18.1
Romantic Poetry	93.8	40.2	Ours ($p = 0.0$)	55.0	63.8	58.9	33.5	**17.2**
			Ours ($p = 0.6$)	62.4	60.3	51.8	35.6	16.2
			Ours ($p = 0.9$)	69.8	55.3	40.3	**36.8**	13.0
Shakespeare	86.1	59.9	Ours ($p = 0.0$)	36.8	65.5	76.9	21.7	15.4
			Ours ($p = 0.6$)	52.1	58.6	65.4	28.2	**16.6**
			Ours ($p = 0.9$)	63.7	48.9	44.2	**29.3**	11.3
Switchboard	99.7	63.1	Ours ($p = 0.0$)	62.9	67.4	77.0	40.8	32.0
			Ours ($p = 0.6$)	77.2	63.7	64.2	**47.5**	**30.2**
			Ours ($p = 0.9$)	84.9	56.6	44.0	46.8	20.1
Overall	88.0	71.9	Ours ($p = 0.0$)	40.1	67.4	78.4	25.3	19.6
			Ours ($p = 0.6$)	48.4	63.4	71.1	28.6	**20.7**
			Ours ($p = 0.9$)	55.7	56.5	56.2	**29.4**	15.8

Table 15: A detailed performance breakup when transferring **to** each style in CDS from the other 10 styles. We test three nucleus sampling (Holtzman et al., 2020) strategies with our trained model by varying the p value between 0.0 (greedy) and 1.0 (full sampling). For reference, the classification accuracy (Orig. ACC) and fluency (Orig. FL) of original sentences in the target style corpus are provided.

Split	Diverse Paraphraser			Non-Diverse Paraphraser		
	Similarity (↑)	Lexical (↓)	Syntactic (↓)	Similarity (↑)	Lexical (↓)	Syntactic (↓)
AAE Tweets	65.1	44.7	0.43	74.3	66.4	0.82
Bible	74.6	48.5	0.55	88.3	73.5	0.92
COHA 1810s-1820s	74.0	50.6	0.51	86.3	71.8	0.92
COHA 1890s-1900s	75.3	52.0	0.50	88.2	75.3	0.93
COHA 1990s-2000s	77.6	57.4	0.53	89.9	80.7	0.95
English Tweets	73.1	52.4	0.50	82.8	75.7	0.91
James Joyce	71.5	47.8	0.35	82.4	69.8	0.82
Lyrics	74.5	52.8	0.52	86.7	78.6	0.92
Romantic Poetry	72.3	46.3	0.44	81.3	67.1	0.86
Shakespeare	67.9	38.7	0.23	81.4	63.4	0.75
Switchboard	71.6	50.1	0.55	81.1	72.4	0.90
Overall	72.5	49.2	0.46	83.9	72.3	0.88

Table 16: A detailed style-wise breakup of the **diverse paraphrase quality** in CDS (the training data for the inverse paraphrase model, described in Section 2.1 and Section 2.4). The ideal paraphraser should score lower on "Lexical" and "Syntactic" overlap and high on "Similiarity". Overall, our method achieves high diversity (51% unigram change and 27% word shuffling, compared to 28% unigram and 6% shuffling for non-diverse), while maintaining good semantic similarity (SIM= 72.5 vs 83.9 for non-diverse) even in complex stylistic settings. We measure lexical overlap in terms of unigram F1 overlap using the evaluation scripts from Rajpurkar et al. (2016). Syntactic overlap is measured using Kendall's τ_B (Kendall, 1938) of shared vocabulary. A $\tau_B = 1.0$ indicates no shuffling whereas a value of $\tau_B = -1.0$ indicates 100% shuffling (complete reversal). Finally, the SIM model from Wieting et al. (2019) is used for measuring similarity.