# Audience-Centric Natural Language Generation via Style Infusion

Samraj Moorjani[1], Adit Krishnan[1], Hari Sundaram[1], Ewa Maslowska[1], Aravind Sankar[1]
[1]University of Illinois at Urbana-Champaign, USA
{samrajm2, aditk2, hs1, ehm, asankar3}@illinois.edu

## Abstract

Adopting contextually appropriate, audience-tailored linguistic styles is critical to the success of user-centric language generation systems (*e.g.,* chatbots, computer-aided writing, dialog systems). While existing approaches demonstrate textual style transfer with large volumes of parallel or non-parallel data, we argue that grounding style on audience-independent external factors is innately limiting for two reasons. First, it is difficult to collect large volumes of audience-specific stylistic data. Second, some stylistic objectives (*e.g.,* persuasiveness, memorability, empathy) are hard to define without audience feedback.

In this paper, we propose the novel task of *style infusion* - infusing the stylistic preferences of audiences in pretrained language generation models. Since humans are better at pairwise comparisons than direct scoring - *i.e., is Sample-A more persuasive/polite/empathic than Sample-B* - we leverage limited pairwise human judgments to bootstrap a style analysis model and augment our seed set of judgments. We then infuse the learned textual style in a GPT-2 based text generator while balancing fluency and style adoption. With quantitative and qualitative assessments, we show that our infusion approach can generate compelling stylized examples with generic text prompts. The code and data are accessible at https://github.com/CrowdDynamicsLab/StyleInfusion.

## 1 Introduction

In this paper, we develop a novel approach to infuse audience-centric styles into pretrained language generation (NLG) models. Learning to synthesize subjective styles is crucial to various applications. For instance, persuasion and memorability in computational advertising and marketing (van Noort et al., 2020). User-centric applications of language generation, such as writing aids, chatbots, and dialog systems, often require these stylistic adjustments depending on both the audience and the task. Prior work often defines textual style with large static sentence collections. However, stylistic objectives such as persuasiveness, memorability, and empathy are hard to define without a target audience (Bell, 1984) due to non-uniform stylistic expectations across diverse user groups. Thus, we suggest that subjective text styles and traits must be defined by the target audience instead of audience-independent data. Our work focuses on two resulting challenges - first, how to collect target audience feedback, and second, how to leverage the limited feedback efficiently for style infusion.

Textual styles - *i.e.,* different linguistic presentations of the same conceptual content - play an integral role in persuasive/memorable communication. For instance, an informal style is less persuasive in formal settings (Kim et al., 2019). The style problem extends across diverse domains, from empathic styling in mental health (Cameron et al., 2018) to fact-driven, simplistic styling in tech support (Okuda and Shoda, 2018). Existing work in textual style transfer (TST) takes two general approaches. The strictly supervised approaches leverage fixed parallel corpora, analogous to machine translation (Hu et al., 2017b), while semi-supervised and unsupervised techniques leverage non-parallel collections of stylized sentences (Shen et al., 2017). Predefined metrics, heuristics, external oracles, and hybrid approaches have also been considered (Jain et al., 2019; Jin et al., 2019a).

Constructing audience-centric or time-evolving / adaptive methods for style transfer remains an open challenge. Existing approaches are guided by rigid modeling considerations and the distributions of fixed style-specific corpora. This is innately limiting for stylistic objectives such as persuasiveness, a trait with a widely disputed definition in existing literature, and multiple external confounds such as preexisting biases and independent features of the persuader (*e.g.,* how many followers they have)

(Al Khatib et al., 2020; Moran et al., 2016; Lowrey, 1998; Berger and Milkman, 2012; Murphy, 2001). Furthermore, it is infeasible to collect extensive annotated collections of text for each audience, style, and application (Pennebaker and King, 1999).

Unlike prior work, we define and incorporate style grounded on our target audience. To address dynamic settings requiring audience-centric linguistic styles, we propose the novel *style-infusion* task. Since human reviewers are better at pairwise style comparisons than direct scoring (Shah et al., 2014), we formulate *style infusion* as follows: how do we *infuse* the stylistic preferences of our audience, via pairwise sentence comparisons, in a generative language model (LM)? Unlike conventional style transfer, our task leverages domain and audience-specific feedback instead of parallel non-parallel sentence collections rendered in any specific style. Further, we adopt an incremental training approach rather than retraining models from scratch.

We bootstrap an initial style analysis model to discriminate the positive and negative samples from audience feedback. Our model then selects additional samples from a generic topical sentence collection to expand the seed set of audience judgments. By separating style analysis and text generation models, we create an adversarial setup to infuse the audience's stylistic feedback in any generative LM. We weight the noisy reward from the style analysis model (discriminator) with a reconstruction loss to balance style adoption and fluency.

In summary, our contributions are as follows:

1. **Audience-centric Style Infusion**: To our knowledge, we are the first to formulate the task of style infusion to tether the definition of style to the target audience. In contrast, prior work defines style in a purely data-driven manner (Shen et al., 2017; Yang et al., 2018). External data limits the definition of style to the context in which it was collected. We propose a more human-centric approach to text styling through explicit audience feedback via pairwise comparisons.

2. **Decoupling Style**: We decouple the style analysis and language generation models for versatility and simplicity. Prior work often unifies these tasks in a single training setup, thus sacrificing incremental learning and infusion of new stylistic preferences of audiences (Jain et al., 2019; Jin et al., 2019a). We introduce an automatically weighted loss, combining an independent reconstruction loss for generation and discriminator-based loss for style, producing a more robust representation of style than in fused settings.

3. **Automatic Style Evaluation**: To the best of our knowledge, we are the first to automatically evaluate the transfer of memorability/persuasiveness. Existing literature has relied on costly manual evaluation as these two traits are hard-to-define stylistic objectives lacking generative work (Li et al., 2020; Tan et al., 2016; Danescu-Niculescu-Mizil et al., 2012). We introduce a new audience-centric correlation metric using a hierarchical Bayesian model to compute the correlations of linguistic features with audience feedback. We then evaluate our model's generations based on their agreements with these audience correlations.

## 2 Related Work

Prior work has explored "style transfer" in diverse settings ranging from "clickbait" headlines to formalizing text (Jin et al., 2020; Chawla and Yang, 2020; Xu et al., 2019). While strictly supervised approaches show high fidelity to input samples (Hu et al., 2017b; Jhamtani et al., 2017), unsupervised and minimally supervised learning are widely applicable since parallel samples are unavailable (Shen et al., 2017; Yang et al., 2018).

Disentanglement, prototype editing, and pseudo-parallel corpus creation are popular approaches. Prototype editing applies stylistic markers to predefined sentence templates (Guu et al., 2018; Li et al., 2018), disentanglement extracts style independent of the content (Shen et al., 2017; Hu et al., 2017a). Audience-centric feedback may not conform to these rigid hypotheses. First, unconstrained generation allows for freedom in sentence and paragraph-level constructs to define the style (Li et al., 2020). Second, the separability of content and style is harder in specialized domains reliant on domain-specific jargon (Woodward-Kron, 2008) and expressions. Our bootstrapping approach shares some commonalities with pseudo-parallel corpus creation (e.g. aligning sentences from two mono-style corpora) (Jin et al., 2019b; Zhang et al., 2018), but only utilizes a generic topical corpus to expand the audience-generated "seed set" of pairwise judg-

ments. Adversarial training has also been used to quantify style (Yang et al., 2022). Our approach explicitly decouples the style discrimination and generation tasks for modularity and incremental training purposes.

We pick two stylistic objectives that are highly audience-dependent and hard to define objectively - memorability and persuasiveness - to evaluate our approach. Prior work in these styles has been limited to analysis but not generation. Tan et al. (2016) and Li et al. (2020) find linguistic patterns, interaction dynamics, and discourse structure are strong identifiers of persuasive arguments, while convincingness (Habernal and Gurevych, 2016), memorability (Danescu-Niculescu-Mizil et al., 2012) have been better explained by linguistic feature correlation. However, there is a lack of work on unconstrained generation of persuasive and memorable text (Dürr and Gloor, 2021; van Noort et al., 2020). Our approach enables us to bridge some of these specific gaps while maintaining a generalized overall formulation.

## 3 Discriminative Language Model

In this section, we train a BERT-based style discriminator to provide feedback to our generator.

### 3.1 Model Architecture and Training

Our style discriminator (style analysis module) adds a fully connected (FC) layer with dropout to pre-trained BERT (Devlin et al., 2019). We use the 'bert-base-uncased' model (Wolf et al., 2019) (12-layers, 768 dimension). We concatenate with a '<SEP>' token and jointly tokenize the compared pair of sentences. The FC layer generates a single output ($\mathbb{R}^{768} \rightarrow \mathbb{R}$). We threshold the sigmoid of the output at 0.5 to decide the preferred sentence. We train all layers (including BERT) on the pairwise audience feedback (batch size 32, 5 epochs, $\eta = 0.0001$, dropout $= 0.2$). We also train a Siamese BERT architecture (Reimers and Gurevych, 2019) with the same settings but find it to underperform BERT (results in Appendix A).

### 3.2 Pairwise Feedback Datasets

We select one pairwise feedback dataset for both the persuasiveness and memorability tasks to evaluate our approach. The UKPConvArg1 corpus (Habernal and Gurevych, 2016) presents pairs of arguments where human annotators select the more persuasive argument. The authors generate 16,000 argument pairs over 16 distinct, non-overlapping topics. Both arguments in a pair belong to the same topic and argue for the same stance (*i.e.,* parallel pairwise feedback). For memorability, we leverage the Cornell Movie-Quotes Corpus (Danescu-Niculescu-Mizil et al., 2012), containing 2,200 paired movie quotes with crowdsourced memorability annotations.

### 3.3 Observations and Validation

Our discriminator achieves 89% accuracy over 5-fold cross-validation for the persuasiveness task. We further validate for overfitting by holding out two topics from the test set and training on the remaining topics, ensuring the discriminator has no exposure to these held-out topics during training. After training from scratch, the discriminator still achieves 87% accuracy on the held-out topics. On the Cornell Movie-Quotes corpus, the discriminator achieves 80% accuracy. We repeat the held-out topic test to validate the classification performance for the memorability task.

In summary, these tests validate the ability of our style discriminator to learn audience style preferences with small volumes of pairwise feedback. In Section 4, we describe our approach to infuse the style discriminator feedback into a generative language model.

## 4 Style-Aware Language Generation

In this section, we infuse the stylistic preferences learned by our style discriminator in Section 3 into a GPT-2 model (Radford et al., 2019) pretrained on the causal language modeling (CLM) objective [1]. The model takes in a textual prompt and generates text, $y$, that we want to infuse with the audience preferred style. For the persuasiveness task, the UKPConvArg1 dataset provides prompts for each argument pair. For the memorability task, we use the previous sentence as the prompt.

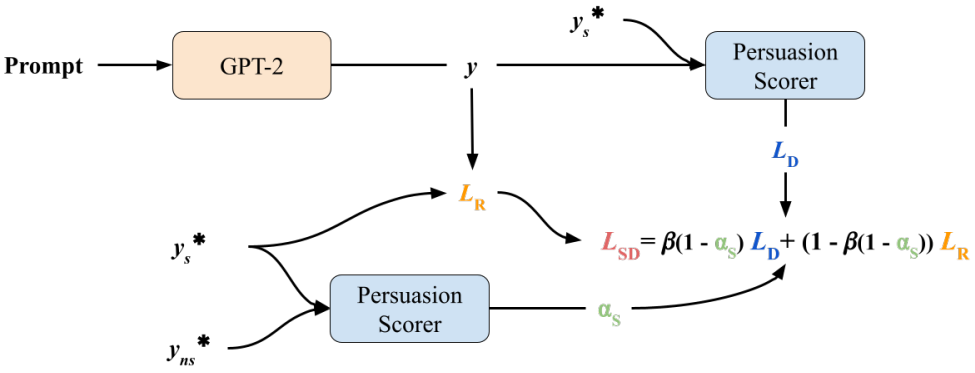
During training, we feed the prompt and feedback pair to GPT-2 - the more preferred (styled) sample, $y_s^*$, and the corresponding less-preferred (non-styled) sample, $y_{ns}^*$. We use an adversarial training paradigm to enable the generator to learn from the discriminator, illustrated in Figure 1.

### 4.1 Training

We utilize two losses during training: a reconstruction loss, $L_R$, and a discriminator loss, $L_D$. The

[1] https://huggingface.co/gpt2

Figure 1: Training diagram that shows how the loss is calculated as a weighted sum of the discriminator ($L_D$) and reconstruction ($L_R$) loss. $\alpha_S$ is decided by the discriminator as a form of contrastive learning.




reconstruction loss is meant to maximize the log probability of the styled training argument, $y_s^*$.

$$L_R = -\frac{1}{N}\sum_{i=1}^{N} \log p(y_s^{*(i)}) \quad (1)$$

The reconstruction loss teaches the model to mimic the gold-standard samples. The discriminator loss is meant to maximize the score of the discriminator, $\mathcal{D}$, and is formulated as:

$$L_D = \mathcal{D}(y_s^*, y) - \frac{1}{N}\sum_{i=1}^{N} \hat{R}_i \log p(y^{(i)}) \quad (2)$$

where $y^{(i)}$ is the $i$-th token of the generated sentence $y$, and $\hat{R}_i$ is a baseline reward meant to reduce the noise from the discriminator. We elaborate on the baseline reward in Appendix B.

We find that too strong of a discriminator loss negatively impacts fluency. Thus, we introduce a regularization constant, $\beta$, to ensure that the discriminator loss remains only a fraction of the loss. The two losses are weighted together to create the final loss as follows:

$$L_{SD} = C \cdot L_D + (1 - C) \cdot L_R \quad (3)$$

where $C = \beta(1 - \alpha_S)$ and $\alpha_S = \mathcal{D}(y_s^*, y_{ns}^*)$. Note that $y_{ns}^*$ is the non-styled training argument.

Instead of making the weighted ratio between the two losses constant, we make them sample dependent. The intuition is that when $\alpha_S$ is high (e.g. the sample is persuasive), we can just use the reconstruction loss to replicate the gold standard which will directly reflect the style. However, when $\alpha_S$ is low (i.e. we have a weak sample), we instead switch to learning the trends from the discriminator. This loss is referred to as the sample-dependent discriminator (SD) loss. We also compare the discriminator loss with a simpler supervised loss defined as $L_S = \frac{1}{N}\sum_{i=1}^{N} \mathcal{D}(y_s^*, y^{(1..i)})$.

### 4.2 Dataset Augmentation Approach

The UKPConv1 and Cornell Movie Quotes corpora we presented in Section 3.2 provide approximately 16,000 and 2,200 unique pairs for stylistic feedback; not nearly enough to train a large language model. To increase our model's breadth of knowledge, we generate additional pairwise feedback with the CNN/Daily Mail dataset (See et al., 2017), containing over 300,000 unique news articles.

First, we generate the Universal Sentence Embeddings (Cer et al., 2018) of all unique sentences in our style corpora (UKPConv1, Cornell) and external corpora (CNN/Daily Mail). For each candidate sentence in the external dataset, $s_i$, we find the top-k similar sentences ($y_1 ... y_k$) in the style corpus to be augmented. We then perform pairwise comparisons $\vee_j \mathcal{D}(s_i, y_j) > 0.5, j \in \{1, ..., k\}$ where if the discriminator prefers the candidate external sentence ($s_i$) over *any one* of the similar sentences ($y_1 ... y_k$) from the style corpus, we include the pair. Through this bootstrapped augmentation method, we ensure we have sentences that are relatively more "styled", as defined by our discriminator, and similar to those in our existing corpus.

### 4.3 Style-Aware Generation with GPT-2

The OpenAI GPT-2 (Radford et al., 2019) model is a large transformer-based language model pretrained on nearly 8 million web pages, allowing generalization to many domains and tasks. This

is closer to the unconstrained scenarios that we wish to target with our style-infusion task. Alternate generators such as pointer-generators rely on copying (Xu et al., 2019), thus introducing more limitations in the extent of style infusion. The ability to extensively pretrain transformer-based models makes them more widely applicable for syle infusion (Gururangan et al., 2020).

In this section, we introduced the adversarial training mechanism for the style-aware language generator and the bootstrapped data augmentation method used to produce robust generations. Next, we will introduce the baselines, evaluation metrics, and training settings.

## 5 Experimental Settings

We compare our architecture against a few strong baselines:

***Pretrained GPT-2*** (Radford et al., 2019) We use this pre-trained model as a representation of average text, allowing us to show shifts in style that occur due to training.

***Fine-tuning*** We fine-tune the pre-trained GPT-2 model using the reconstruction objective on the style-specific corpus only (e.g. UKPConv1).

***Fine-tuning + Data Augmentation*** We fine-tune the pre-trained GPT-2 model using the reconstruction objective on the augmented data.

***TitleStylist*** (Jin et al., 2020) We adapt the stylistic headline generation framework to generate stylistic text based on a prompt. Jin et al. (2020) utilize a Denoising Autoencoder with parameter sharing to disentangle style from content to control the style with a set of parameters.

**Training Settings** For all GPT-2 based models, we use a base GPT-2 model from Huggingface (Wolf et al., 2019) (1024 dimensions, Adam optimizer, $\eta = 5e-5$). Because of the length of our text and size of our models, we utilize DeepSpeed (Rasley et al., 2020) to distribute training over two 32GB V100s, and we train with FP16 mixed precision. We experiment with the loss parameters of $C$ and $\beta$ and discuss our findings in section 8.

**Evaluation Metrics** We take a deeper look into the annotator labels in the UKPConvArg1 dataset and we find that some linguistic features play a significant role in the persuasiveness of text.

We create a hierarchical Bayesian model to find the correlation between a set of collected linguistic features and the desired style. We first take the unique sentences from a dataset and compute a set of linguistic features over them. A full list of features can be found in Appendix C.

For each linguistic feature-topic pair, we infer the correlation between the feature and the text that demonstrates the style by running a Markov Chain Monte-Carlo (MCMC) process using the No-U-Turn Sampler (NUTS). We elaborate on the calculations in Appendix C.2. Note that the results we show are in the logit scale, meaning even a change of $\mp 1$ has a big effect on the probability (about a 23% difference in odds of winning).

The models then generate text based on the prompts in a held-out test set and we calculate the features of the generations. We run a t-test to determine if the difference in features between a pretrained GPT-2 and one of our models is statistically significant. This evaluation shows how our trained model learns to use these linguistic features to construct more stylized arguments.

In addition, we use *pyrouge* library to collect the ROUGE (Lin, 2004) score, a commonly used metric that measures the N-gram overlap between the training and generated arguments. While these scores will not tell us how persuasive our generations are, they will ensure that the generations remain on topic.

Lastly, we compute the BERTScore (Zhang et al., 2019), another automatic evaluation metric that computes token similarity using contextual embeddings. The BERTScore represents the semantic similarity of the generations to the test set which will ensure generations are relevant, but not necessarily persuasive.

## 6 Results on Persuasiveness

In this section, we analyze our results by showing a significant usage of linguistic features that resemble persuasive text, showing generated text, and with standard metrics.

### 6.1 Linguistic Feature Correlations

Figure 2 shows the correlations between linguistic features and convincingness in the UKPConvArg1 corpus. The model details are in Appendix C.

We find a strong positive correlation between readability and winning arguments. This is reflected by both readability scores (*e.g.,* SMOG, Flesch-Kincaid, etc.) and correlation with smaller words, fewer total dependencies, and a smaller overall total dependency distance. We notice a positive correlation with speed and volume. Toubia

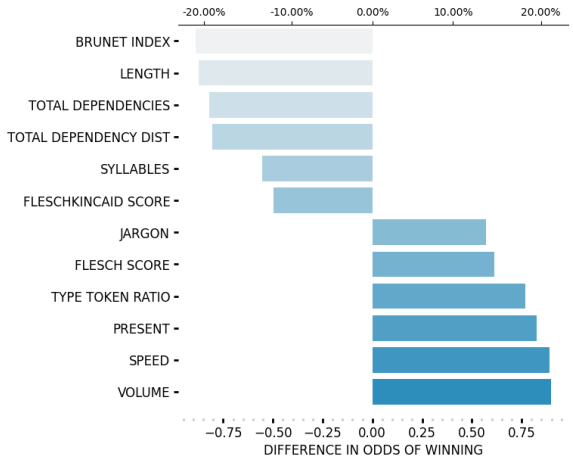



Figure 2: The correlations between linguistic features and convincingness in the UKPConvArg1 corpus. The lower x-axis is in the logit scale, and the percentage difference in odds of winning is on the upper x-axis. The figure is read as: if the feature for argument A is one standard deviation greater than the feature for argument B, the odds of A winning shift by the respective percent value. Notice that the correlations show a strong positive correlation with readability (*e.g.,* the Flesh score is positive while length and average syllables are negative).

et al. (2021) define speed as the total distance covered by a text's word embeddings, normalized by the length of the text. Volume represents the amount of material covered by the text, calculated by estimating the volume enclosed by the word embeddings (Toubia et al., 2021). We also find a negative correlation with passive voice and a positive correlation with misspelled words (not shown for brevity).

We run a significance test to see how well our models learned the style (see Appendix C.3). Most models consistently learn pronounced trends (*i.e.,* Brunet index, length, speed, and volume). The augmented data likely led to this change because fine-tuning on the augmented set displays the same trends. In cases like total dependencies (TD) and the ratio of present tense verbs, models trained with the sample-dependent discriminator (SD) loss are significantly better at learning the trend, despite the data not actively showing the trend (or completely opposing it). In the case of Flesch score, models trained with SD loss can nullify the trend which occurs in the incorrect direction. This displays that models trained with the SD loss are substantially better at learning from the dataset than the baselines and models trained with the sample-dependent supervised (SS) loss. One example of failure is the ratio of jargon, likely because the model could not generate out of vocabulary words, but this is a limitation of how we define jargon.

### 6.2 Sample Generations

Table 1: The baseline models and our model (GPT2 with the SD loss and $\beta = 0.5$ on the augmented data). The models are fed the input "Plastic bottles are bad because" and "Gaming is good for child development because", generating up to 100 tokens using beam search.

| Model & Generated Text |
|---|
| Example 1: |
| **GPT2**: Plastic water bottles are bad because they are not recyclable," he said. "They are not good for the environment, they are not recyclable." |
| **GPT2-Aug**: Plastic water bottles are bad because it causes pollution. Also water bottles also cost more than regular water. |
| **TStylist**: Plastic water bottles are bad because plastic water bottles are bad for our environment. |
| **Ours (SD-0.5)**: Plastic water bottles are bad because they are not recycled and end up in landfills which lead to pollution of our environment. Plastic requires up to 47 million gallons of oil per year to produce. |
| Example 2: |
| **GPT2**: Gaming is good for child development because it teaches them how to use a computer. It's good for the economy, it's good for the environment, it's good for the children, it's ... |
| **GPT2-Aug**: Gaming is good for child development because it allows the child to grow and develop. I believe eSports (LoL) should be a spectator sport and not a major part of the sports calendar. |
| **TStylist**: Gaming is good for child development because gaming is good for your child. |
| **Ours (SD-0.5)**: Gaming is good for child development because it allows children to grow up in a world where they are exposed to a wide variety of ideas and experiences. |

Table 1 shows the generations of three baselines and our best-performing method. We find that for both prompts, the generations of models trained with the sample-dependent discriminator (SD) loss generally have the highest values of speed, volume, and lexical diversity. For the second prompt, the speed and volume of our generation are larger than that of GPT2 and TStylist, but slightly smaller than that of GPT2 fine-tuned on the augmented data. Intuitively, this makes sense because the GPT-Aug generation covers much more information in the same time frame; however, this information isn't relevant to the argument, making our generation much more sensible. The baselines often suffer from neural degeneration, but the model trained with the SD loss does not face this issue. Since length had a strong negative correlation with per-

suasiveness, the model likely implicitly learned from the discriminator to handle this kind of neural degeneration. However, it is still an issue in some cases with out-of-domain samples.

Table 2: ROUGE-{1,2, L} scores and BERT scores (F1) for all models. Baseline models: GPT2, GPT-2 fine-tuned on UKPConvArg1, GPT-2 with augmented data, TitleStylist (Jin et al., 2020). Our models are trained on augmented data and a sample-dependent discriminator (SD) or sample-dependent supervised (SS) loss with parameter $\beta$. The baseline ROUGE score increases due to data augmentation; the relevance of our models' generations is largely insensitive to loss type and parameter value.

| MODEL | RG-1 | RG-2 | RG-L | B-F1 |
|---|---|---|---|---|
| GPT2 | 0.1856 | 0.0968 | 0.1769 | 89.28 |
| GPT2-UKP | 0.2474 | 0.1061 | 0.1989 | 87.21 |
| GPT2-Aug | 0.2987 | 0.1845 | 0.2774 | 86.90 |
| TStylist | 0.2578 | 0.1569 | 0.2391 | 84.70 |
| SS-0.1 | 0.2925 | 0.1802 | 0.2717 | 88.95 |
| SS-1.0 | 0.2862 | 0.1774 | 0.2634 | 89.18 |
| SD-0.1 | 0.3036 | 0.1903 | 0.2802 | 88.75 |
| SD-0.5 | **0.3168** | **0.2296** | **0.3065** | **89.56** |
| SD-0.8 | 0.2872 | 0.1957 | 0.2733 | 89.12 |
| SD-1.0 | 0.2929 | 0.2224 | 0.2848 | 89.05 |

### 6.3 Automatic Metrics

We compare the ROUGE scores of our experimental models in Table 2, ensuring that the topics in the test set are not discussed anywhere in the UKP-ConvArg1 or augmented datasets. The data augmentation leads to a sharp increase in the ROUGE scores of the generations, showing that it is essential for robust and relevant generations. The results are relatively insensitive to variation in $\beta$ parameter that controls the tradeoff between reconstruction loss ($L_R$) and discriminator loss ($L_R$). These scores show that our models generate relevant, but not necessarily persuasive, text. We find similar insights from the BERTScore; although the augmentation has a slight negative impact on the score, the difference is negligible.

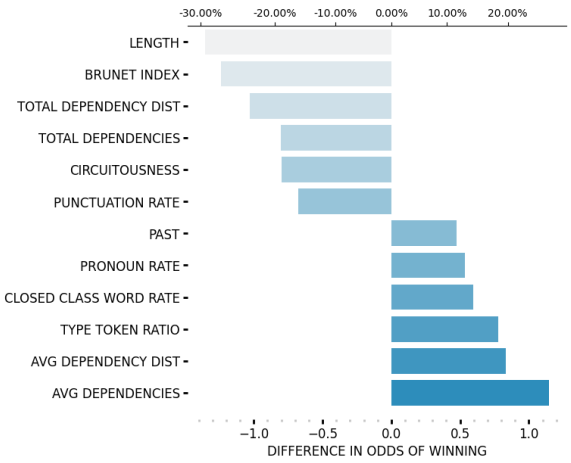
## 7 Results on Memorability

In this section, we focus on memorability and show that our model can generate more robust, relevant, and memorable text than the baselines.

### 7.1 Linguistic Feature Correlations

We train a Bayesian hierarchical model for the Cornell Movie Quotes corpus, which produces the correlations shown in Figure 3. We find a strong negative correlation with long, winding text, shown by the trends in total dependencies, total dependency distance, length, and circuitousness. A higher circuitousness implies that a less direct route was taken to convey information (Toubia et al., 2021). Circuitousness is detrimental to memorability as winding text tends to be harder to remember. The negative correlation with the punctuation rate and positive correlation with the average dependencies show that more memorable text tends to have a few sentences, independent of the length of sentences. Lastly, there is a strong emphasis on uncommon vocabulary with a negative correlation with the Brunet index and a positive correlation with token type ratio. This is supported by findings from (Danescu-Niculescu-Mizil et al., 2012) who find that memorable quotes are built upon less common word choices.

Figure 3: The correlations between linguistic features and persuasiveness in the Cornell Movie-Quotes Corpus. The lower x-axis is in the logit scale, and the percentage difference in odds of winning is on the upper x-axis. Notice that the correlations show a negative correlation with long and winding text (*i.e.,* circuitousness (Toubia et al., 2021)).




We run the significance test on a held-out test set to see how well our models learned to generate memorable text. In some cases in Table 8, models trained with the sample-dependent discriminator (SD) loss have similar performance as the fine-tuned models, indicating that some relevant features are learned solely from fine-tuning. However, many other incorrect trends are corrected with training using the SD loss. The only feature that does not improve is the pronoun rate, likely because of shorter sentences with more emphasis on uncommon word choices.

### 7.2 Sample Generations

We look at a few examples of generations to see how training influenced the model's generations in Table 3. There is a visible decrease in the number of sentences, overall length, and circuitousness in both the GPT2-Aug and SD-0.5 models. This reflects the trends shown in Table 8, but we can see that the model trained with the SD loss generates sentences that are more sophisticated than the fine-tuned GPT2 model's. TitleStylist generates sentences that are identical to our generation feature-wise but are not as sophisticated as our generations.

| MODEL & GENERATED TEXT |
|---|
| EXAMPLE 1: |
| **GPT2**: The more you know about it, the more likely you are to believe that it is true. If you don't believe that it is true, you're not alone. If you don't believe that it is true, you're not alone... |
| **GPT2-Aug**: The more you smoke, the more you will smoke |
| **TStylist**: The more you need to know. |
| **Ours (SD-0.5)**: The more you learn, the more you see. |
| EXAMPLE 2: |
| **GPT2**: When solving problems, dig a hole in the ground and dig a hole in the ground and dig a hole in the ground and dig a hole in the ground and ... |
| **GPT2-Aug**: When solving problems, dig your heels in and try to find a solution, even if you don't have the answers, and even if you don't know the answers. |
| **TStylist**: When solving problems, dig better and better. |
| **Ours (SD-0.5)**: When solving problems, dig deeper than a grave. |

Table 3: Generations of GPT2, GPT2 fine-tuned on augmented data, and GPT2 with the SD loss ($\beta = 0.5$) on the augmented data. The models are fed in the inputs "The more you" and "When solving problems, dig".

### 7.3 Automatic Metrics

Similar to persuasiveness, Table 4 shows that the ROUGE scores increase mainly due to data augmentation. Once again, these results demonstrate that the augmented data leads to more relevant generations, increasing the breadth of knowledge transferred to the model. The same trends generally hold for the BERTScore, which shows that the generations remain semantically relevant.

We show that our model generates more robust, relevant, and memorable text than the baselines. Next, we discuss how tuning the loss parameters affects generations.

Table 4: ROUGE-{1,2, L} scores and BERT scores (F1) for all models. Baseline models: GPT2, GPT-2 fine-tuned on UKPConvArg1, GPT-2 with augmented data, TitleStylist (Jin et al., 2020). Our models are trained on augmented data, and a sample-dependent discriminator (SD) or sample-dependent supervised (SS) loss with parameter $\beta$. The baseline ROUGE score increases due to data augmentation; again, the relevance of generations is largely independent of loss type and parameter value.

| MODEL | RG-1 | RG-2 | RG-L | B-F1 |
|---|---|---|---|---|
| GPT2 | 0.1503 | 0.0853 | 0.1461 | 81.12 |
| GPT2-IMDB | 0.1579 | 0.0853 | 0.1510 | **88.87** |
| GPT2-Aug | 0.2737 | 0.1703 | 0.2685 | 87.24 |
| TStylist | 0.2542 | 0.1617 | 0.2439 | 85.99 |
| SS-0.1 | 0.2746 | **0.1759** | 0.2668 | 85.98 |
| SS-1.0 | 0.2740 | 0.1723 | 0.2661 | 85.93 |
| SD-0.1 | 0.2743 | 0.1735 | 0.2686 | 86.87 |
| SD-0.5 | 0.2718 | 0.1706 | 0.2680 | 83.94 |
| SD-0.8 | **0.2812** | 0.1705 | **0.2733** | 85.81 |
| SD-1.0 | 0.2739 | 0.1681 | 0.2675 | 86.12 |

Table 5: Generations of GPT2 trained with the sample-dependent discriminator loss objective with different values of $\beta$. The generations for SD-0.5 and SD-1.0 tend to be much better than for SD-0.1

| MODEL & GENERATED TEXT |
|---|
| **SD-0.1:** IE sucks and makes development on your computer much more difficult than it should be. I believe that Internet Explorer (IE) is far inferior to Internet Explorer (IE) and Internet Explorer (IE) is far inferior to Internet Explorer (IE) |
| **SD-0.5:** IE sucks and makes development more difficult |
| **SD-1.0:** IE sucks and makes development on your computer much more difficult than it should be. |

## 8 Empirical Observations

We analyze how the value of $\beta$ affects generations, finding that generations from $\beta = 0.1$ suffer the same degeneration as fine-tuning while higher values avoid these issues. Because $\alpha_S$ is not always 1, the constant in front of the discriminator loss is less than $\beta$. Consequently, the discriminator is not given enough weight, and the generator cannot learn as effectively from the discriminator. It is difficult to distinguish differences between $\beta = 0.5$ and $\beta = 1.0$, but aside from $\beta = 0.5$, $\beta = 1.0$ outperforms every other value of $\beta$.

We also experiment with hard-coding the coefficients for the discriminator and reconstruction loss in Table 6. Putting too much weight on the discriminator loss, $L_D$, (*i.e.,* 0.9) leads to poor quality arguments having some of the strongest linguistic feature changes (*e.g.,* shorter length). Conversely, limiting $L_D$ to 0.1 leads to much stronger genera-

Table 6: Generations of our model trained with a mixed reconstruction and discriminator loss objective with hard-coded weights (as opposed to sample-dependent).

| MODEL & GENERATED TEXT |
|---|
| EXAMPLE 1: |
| **0.9 Supervised + 0.1 MLE:** Schools should teach physical education because it's a good thing. |
| **0.1 Supervised + 0.9 MLE:** Schools should teach physical education because PE helps children develop good habits later on in life. Plus, there's the benefit of working together as a team that doesn't always happen in other classes. |
| EXAMPLE 2: |
| **0.9 Supervised + 0.1 MLE:** Plastic water bottles are bad because they are not recyclable. |
| **0.1 Supervised + 0.9 MLE:** Plastic water bottles are bad because they are bad for the environment and they are bad for the economy.Some people think that bottled water is bad for consumers and should only be used in situations such as disasters when no other clean water is available. |
| EXAMPLE 3: |
| **0.9 Supervised + 0.1 MLE:** Gaming is good for child development because you can play with other kids. |
| **0.1 Supervised + 0.9 MLE:** Gaming is good for child development because it teaches them how to think and solve problems. It also teaches them how to communicate with each other. |

tions. We introduced the $\beta$ parameter to cap $L_D$ at $\beta$. Because of the $\beta$ parameter, the previous experiments show similar but less obvious trends.

## 9 Conclusion

In this paper, we introduced *style infusion* to motivate infusing audience-centric, stylistic preferences into unconstrained natural language generation models. We present a bootstrapped data augmentation method for limited pair-wise audience feedback and an adversarial training framework with a decoupling loss to train a style-infused GPT-2. Through an automatic evaluation method for the transfer of audience-specific styles, we show that our approach generates compelling stylized examples with generic text prompts better than the baselines.

Synthesizing text with subjective styles, such as persuasion and memorability, remains a significant challenge in domains like computational advertising. Our work takes the first few steps to address this problem. We plan to continue improving our work in many directions, such as incorporating long-document attention mechanisms (Beltagy et al., 2020) to capture document-level style features and altering the discourse structure to convey information in a more interpretable manner.

## 10 Limitations

As with other unconstrained natural language generation applications, our system is prone to issues like degeneration from beam search and neural hallucinations. To combat the former, we post-process generations, but future work will hopefully provide better methods to prevent this issue. For the latter, we increase our dataset with samples from the CNN/DM dataset, partially mitigating the problem, but out-of-domain topics still suffer. Increasing the dataset size will only work for so long due to diminishing marginal returns.

Due to the limited amount of data available, we considered iteratively training the discriminator with the augmented data while we trained the generator. Ultimately, we felt that the weak labels would dilute the learned trends in the discriminator, but it may be interesting to see how it affects the performance of the framework. Currently, collecting pairwise datasets to use with this framework can be viewed as a limitation. With increasing interest in the computational synthesis of persuasive text and imagery, we expect to see more relevant curated datasets in the near future. Generating pairwise data through human subject experiments is expensive, which is why the data augmentation methods introduced in this paper are crucial for future work.

We also note that our framework is limited by the computational resources available to us. Thus, we were unable to effectively support long text generation while preserving the quality of the generated text. During training, we decrease the batch size and utilize the DeepSpeed framework (Rasley et al., 2020), but it is still insufficient to handle long text. Furthermore, traditional left-to-right generation struggles with long text as the topics tend to diverge. Because many styles, like persuasiveness, are dependent on paragraph-level features in addition to sentence-level ones, it is beneficial for our application to support longer texts.

Lastly, one of the biggest limitations of this paper is in showing the effectiveness of the architecture we choose. Because most baselines are in style transfer and fundamentally differ from our task, we find it difficult to make a fair comparison with prior work. Regardless, style infusion is a critical step for unconstrained NLG systems such as dialogue systems and chatbots, especially in the context of human-centric stylistic objectives, which are already difficult enough to define.

## 11 Ethics Statement and Broader Impact

Our objective for developing a stylistic generative language model that leverages domain and audience-specific feedback is to enable unconstrained generation applications to appeal to more human users. For example, generating more persuasive real news might help combat misinformation by propagating the truth faster than falsehoods. In advertising and communication, persuasiveness and memorability are critical traits and having an unconstrained generation model that could replicate these features would have a multitude of positive applications, especially in targeted interventions. Previous research has mostly focused predicting audience characteristics and targeting, but not on synthesizing matching messages.

We acknowledge the dual-use concerns of the misuse of such a generation framework to, for example, spread misinformation. For this reason, we do not release the model or the pretrained generator checkpoint used in this work.

## Acknowledgements

This work used the Extreme Science and Engineering Discovery Environment (XSEDE) Expanse GPU cluster, which is supported by National Science Foundation grant number ACI-1548562 (Towns et al., 2014). This work was also supported by the National Center for Supercomputing Application's Nano cluster.

## References

Khalid Al Khatib, Michael Völske, Shahbaz Syed, Nikolay Kolyada, and Benno Stein. 2020. Exploiting personal characteristics of debaters for predicting persuasiveness. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7067–7072, Online. Association for Computational Linguistics.

Allan Bell. 1984. Language style as audience design. *Language in Society*, 13(2):145–204.

Iz Beltagy, Matthew E Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*.

Jonah Berger and Katherine L. Milkman. 2012. What makes online content viral? *Journal of Marketing Research*, 49(2):192–205.

Gillian Cameron, David Cameron, Gavin Megaw, Raymond Bond, Maurice Mulvenna, Siobhan O'Neill, Cherie Armour, and Michael McTear. 2018. Assessing the usability of a chatbot for mental health care. In *International Conference on Internet Science*, pages 121–132. Springer.

Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St. John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, Yun-Hsuan Sung, Brian Strope, and Ray Kurzweil. 2018. Universal sentence encoder. *CoRR*, abs/1803.11175.

Kunal Chawla and Diyi Yang. 2020. Semi-supervised formality style transfer using language model discriminator and mutual information maximization. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2340–2354, Online. Association for Computational Linguistics.

Cristian Danescu-Niculescu-Mizil, Justin Cheng, Jon Kleinberg, and Lillian Lee. 2012. You had me at hello: How phrasing affects memorability. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 892–901, Jeju Island, Korea. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Sebastian Dürr and Peter A. Gloor. 2021. Persuasive natural language generation - A literature review. volume abs/2101.05786.

Suchin Gururangan, Ana Marasovic, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. 2020. Don't stop pretraining: Adapt language models to domains and tasks. *CoRR*, abs/2004.10964.

Kelvin Guu, Tatsunori B Hashimoto, Yonatan Oren, and Percy Liang. 2018. Generating sentences by editing prototypes. *Transactions of the Association for Computational Linguistics*, 6:437–450.

Ivan Habernal and Iryna Gurevych. 2016. Which argument is more convincing? analyzing and predicting convincingness of web arguments using bidirectional LSTM. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1589–1599, Berlin, Germany. Association for Computational Linguistics.

Matthew Honnibal and Ines Montani. 2017. spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing. To appear.

Zhiting Hu, Zichao Yang, Xiaodan Liang, Ruslan Salakhutdinov, and Eric P. Xing. 2017a. Controllable text generation. *CoRR*, abs/1703.00955.

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

Parag Jain, Abhijit Mishra, Amar Prakash Azad, and Karthik Sankaranarayanan. 2019. Unsupervised controllable text formalization. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 6554–6561.

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

Di Jin, Zhijing Jin, Joey Tianyi Zhou, Lisa Orii, and Peter Szolovits. 2020. Hooks in the headline: Learning to generate headlines with controlled styles. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 5082–5093. Association for Computational Linguistics.

Zhijing Jin, Di Jin, Jonas Mueller, Nicholas Matthews, and Enrico Santus. 2019a. Imat: Unsupervised text attribute transfer via iterative matching and translation. *arXiv preprint arXiv:1901.11333*.

Zhijing Jin, Di Jin, Jonas Mueller, Nicholas Matthews, and Enrico Santus. 2019b. IMaT: Unsupervised text attribute transfer via iterative matching and translation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3097–3109, Hong Kong, China. Association for Computational Linguistics.

Soomin Kim, Joonhwan Lee, and Gahgene Gweon. 2019. Comparing data from chatbot and web surveys: Effects of platform and conversational style on survey response quality. In *Proceedings of the 2019 CHI conference on human factors in computing systems*, pages 1–12.

Jialu Li, Esin Durmus, and Claire Cardie. 2020. Exploring the role of argument structure in online debate persuasion. *CoRR*, abs/2010.03538.

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.

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

Tina Lowrey. 1998. The effects of syntactic complexity on advertising persuasiveness. *Journal of Consumer Psychology*, 7:187–206.

Meghan Bridgid Moran, Melissa Lucas, Kristen Everhart, Ashley Morgan, and Erin Prickett. 2016. What makes anti-vaccine websites persuasive? a content analysis of techniques used by anti-vaccine websites to engender anti-vaccine sentiment. *Journal of Communication in Healthcare*, 9(3):151–163.

P.Karen Murphy. 2001. What makes a text persuasive? comparing students' and experts' conceptions of persuasiveness. *International Journal of Educational Research*, 35(7):675–698.

Takuma Okuda and Sanae Shoda. 2018. Ai-based chatbot service for financial industry. *Fujitsu Scientific and Technical Journal*, 54(2):4–8.

James W Pennebaker and Laura A King. 1999. Linguistic styles: language use as an individual difference. *Journal of personality and social psychology*, 77(6):1296.

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

Marc'Aurelio Ranzato, Sumit Chopra, Michael Auli, and Wojciech Zaremba. 2015. Sequence level training with recurrent neural networks. *arXiv preprint arXiv:1511.06732*.

Jeff Rasley, Samyam Rajbhandari, Olatunji Ruwase, and Yuxiong He. 2020. Deepspeed: System optimizations enable training deep learning models with over 100 billion parameters. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '20, pages 3505–3506, New York, NY, USA. Association for Computing Machinery.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. *CoRR*, abs/1908.10084.

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1073–1083, Vancouver, Canada. Association for Computational Linguistics.

Nihar B. Shah, Sivaraman Balakrishnan, Joseph Bradley, Abhay Parekh, Kannan Ramchandran, and Martin Wainwright. 2014. When is it better to compare than to score?

Tianxiao Shen, Tao Lei, Regina Barzilay, and Tommi Jaakkola. 2017. Style transfer from non-parallel text by cross-alignment. *Advances in neural information processing systems*, 30.

Abhishek Shivkumar, Jack Weston, Raphael Lenain, and Emil Fristed. 2020. Blabla: Linguistic feature extraction for clinical analysis in multiple languages. *arXiv preprint arXiv:2005.10219*.

Chenhao Tan, Vlad Niculae, Cristian Danescu-Niculescu-Mizil, and Lillian Lee. 2016. Winning arguments: Interaction dynamics and persuasion strategies in good-faith online discussions. *CoRR*, abs/1602.01103.

Olivier Toubia, Jonah Berger, and Jehoshua Eliashberg. 2021. How quantifying the shape of stories predicts their success.

J. Towns, T. Cockerill, M. Dahan, I. Foster, K. Gaither, A. Grimshaw, V. Hazlewood, S. Lathrop, D. Lifka, G. D. Peterson, R. Roskies, J. Scott, and N. Wilkins-Diehr. 2014. Xsede: Accelerating scientific discovery. *Computing in Science & Engineering*, 16(05):62–74.

Guda van Noort, Itai Himelboim, Jolie Martin, and Tom Collinger. 2020. Introducing a model of automated brand-generated content in an era of computational advertising. *Journal of Advertising*, 49(4):411–427.

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

Robyn Woodward-Kron. 2008. More than just jargon–the nature and role of specialist language in learning disciplinary knowledge. *Journal of English for Academic Purposes*, 7(4):234–249.

Peng Xu, Chien-Sheng Wu, Andrea Madotto, and Pascale Fung. 2019. Clickbait? sensational headline generation with auto-tuned reinforcement learning. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3056–3066.

Haitong Yang, Guangyou Zhou, and Tingting He. 2022. Adversarial separation network for text style transfer. *ACM Trans. Asian Low-Resour. Lang. Inf. Process.*, 21(2).

Zichao Yang, Zhiting Hu, Chris Dyer, Eric P Xing, and Taylor Berg-Kirkpatrick. 2018. Unsupervised text style transfer using language models as discriminators. *Advances in Neural Information Processing Systems*, 31.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.

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*.

# A Siamese BERT Discriminator

To validate our results, we tried another architecture, similar to Siamese BERT (Reimers and Gurevych, 2019), where we tokenized the texts individually and passed them through their own BERT layers, producing two embeddings, $e_1$ and $e_2$. We concatenated the two outputs along with the distance between the two as follows:

$$[e_1; e_2; |e_2 - e_1|]$$

We passed this new vector of $\mathbb{R}^{3\times h}$, where $h$ is the hidden dimension of BERT, through a fully connected classification layer.

While the original discriminator achieves approximately 89% accuracy on the random test set, the Siamese BERT model achieves a smaller, but still significant, 83% accuracy. On the Cornell Movie-Quotes corpus, with the same hyperparameters, the original discriminator achieves 80% accuracy and a 77% accuracy with the Siamese BERT architecture. We choose to use the simpler discriminator architecture because it seems to capture style better than the Siamese BERT architecture.

# B Baseline Reward

The baseline reward is meant to reduce the noise from the reward given by our discriminator (Ranzato et al., 2015). The baseline reward, $\hat{R}_i$, is calculated using a linear layer, with the input being the hidden states of our generator at timestep $i$. The intuition is that the linear layer approximates the value of the reward for a certain timestep and in practice, reduces the variance from the reward. We train the linear layer with the following loss:

$$L_{BR} = \frac{1}{N}\sum_{i=1}^{N} |\mathcal{D}(y_s^*, y) - \hat{R}_i|^2 \qquad (4)$$

where $\mathcal{D}(y_s^*, y)$ is the output of the discriminator when fed a gold argument and the generated argument (i.e. the reward).

# C Linguistic Feature Correlation

## C.1 Collected Linguistic Features

We collected the following linguistic features: length, verb tenses (e.g. future, past, etc.), punctuation rates, readability scores (Flesch score, Flesch-Kincaid score, Gunning Fog score, SMOG score, Dale-Chall score), part of speech rates (noun rate,

| Model | BI | Length | TD | TDD | Syllables | Flesch-Kincaid | Jargon | Flesch | TTR | Present | Speed | Volume |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| GPT2 | - | - | - | - | - | - | - | - | - | - | - | - |
| GPT2-16k | ✓ | ✗✗✗✗ | ✗✗✗✗ | - | - | - | ✗ | - | ✓✓ | ✗✗✗✗ | - | ✓✓✓✓ |
| GPT2-Aug | ✓✓✓✓ | ✓✓✓✓ | ✓ | ✓✓✓✓ | - | ✗✗✗✗ | - | ✗✗✗✗ | ✓ | - | ✓✓✓✓ | ✓✓ |
| TStylist | ✓ | ✓✓✓✓ | ✓✓✓ | - | ✓✓ | ✗✗✗✗ | ✗✗✗✗ | ✗✗✗✗ | ✓ | - | ✓✓✓ | ✓✓✓ |
| AP-0.1 | ✓✓✓✓ | ✓✓✓✓ | ✓✓ | ✓✓✓✓ | - | ✗✗✗✗ | - | ✗✗✗✗ | ✓✓ | ✓ | ✓✓ | ✓✓✓ |
| AP-1.0 | ✓✓✓✓ | ✓✓✓✓ | ✓ | ✓✓✓✓ | - | ✗✗✗✗ | - | ✗✗✗✗ | ✓✓ | - | ✓✓✓ | ✓✓✓✓ |
| SD-0.1 | - | - | ✓ | - | ✗✗ | ✗✗✗✗ | - | ✗✗✗✗ | ✗ | - | ✓ | - |
| SD-0.5 | ✓✓✓✓ | ✓✓✓✓ | ✓✓✓✓ | ✓✓✓✓ | - | - | - | ✗ | ✓✓✓✓ | ✓✓✓✓ | ✓✓✓✓ | ✓✓✓✓ |
| SD-0.8 | - | ✓✓✓✓ | ✓✓✓✓ | ✓ | - | ✗✗ | - | ✗✗✗ | ✓✓ | ✓ | ✓✓✓✓ | ✓✓✓✓ |
| SD-1.0 | ✓✓✓✓ | ✓✓✓✓ | ✓✓✓✓ | ✓✓✓ | ✗ | - | - | - | ✓✓✓✓ | ✓✓✓✓ | ✓✓✓✓ | ✓✓✓✓ |

Table 7: Significance tests on the change in features between a pretrained GPT-2 model and a trained model. In order, these features are: Brunet Index (BI), Length (in characters), Total Dependencies (TD), Total Dependency Distance (TDD), Average Syllables per Word, Flesh-Kincaid readability score, Ratio of Jargon (i.e. out of vocab words), Flesch readability score, Token Type Ratio (TTR), Ratio of Present Tense Verbs, Speed, and Volume. In the table, the number of checks or crosses indicates the level of p-value and the correctness of the direction of the trend. Note that ✓: $p < 0.05$, ✓✓: $p < 0.01$, ✓✓✓: $p < 0.001$, ✓✓✓✓: $p < 0.0001$.

| Model | Length | BI | TDD | TD | Circuitousness | PunctRate | Past | Pronoun Rate | CCW Rate | TTR | ADD | AD |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| GPT2 | - | - | - | - | - | - | - | - | - | - | - | - |
| GPT2-IMDB | - | ✓✓✓✓ | ✗✗ | - | ✗ | ✓ | ✓✓✓ | ✗✗✗✗ | ✓✓✓✓ | - | ✓✓✓✓ | ✓✓✓✓ |
| GPT2-Aug | ✓✓✓✓ | ✓✓✓✓ | ✓✓✓✓ | ✓✓✓✓ | ✓✓✓✓ | ✓ | ✓✓✓✓ | ✗✗✗✗ | ✗✗✗✗ | ✓✓✓✓ | ✗✗✗✗ | ✗✗✗✗ |
| TStylist | ✓✓✓✓ | ✓✓✓✓ | ✓✓✓ | ✓✓✓ | ✓✓✓ | ✓ | ✓✓✓✓ | ✗✗✗ | - | ✓✓✓✓ | ✗✗✗ | ✓✓✓✓ |
| AP-0.1 | ✓✓✓✓ | ✓✓✓✓ | ✓✓✓✓ | ✓✓✓✓ | ✓✓✓✓ | - | ✓✓✓✓ | ✗✗✗✗ | ✗✗✗✗ | ✓✓✓✓ | ✗✗✗✗ | ✗✗✗✗ |
| AP-1.0 | ✓✓✓✓ | ✓✓✓✓ | ✓✓✓✓ | ✓✓✓✓ | ✓✓✓✓ | - | ✓✓✓✓ | ✗✗✗✗ | ✗✗✗✗ | ✓✓✓✓ | ✗✗✗✗ | ✗✗✗✗ |
| SD-0.1 | ✓✓✓✓ | ✓✓✓✓ | ✓✓✓✓ | ✓✓✓✓ | ✓✓✓✓ | - | ✓✓✓✓ | ✗✗✗✗ | ✗✗✗ | ✓✓✓✓ | ✗✗✗✗ | ✗✗✗✗ |
| SD-0.5 | ✓✓✓✓ | ✓✓✓✓ | ✓✓✓✓ | ✓✓✓✓ | ✓✓✓✓ | ✓ | ✓✓✓✓ | ✗✗✗ | ✓✓✓✓ | ✓✓✓✓ | ✓✓ | ✓✓✓✓ |
| SD-0.8 | ✓✓✓✓ | ✓✓✓✓ | ✓✓✓✓ | ✓✓✓✓ | ✓✓✓✓ | - | ✓✓✓✓ | ✗✗✗✗ | ✗✗ | ✓✓✓✓ | ✗✗✗✗ | ✗✗✗✗ |
| SD-1.0 | ✓✓✓✓ | ✓✓✓✓ | ✓✓✓✓ | ✓✓✓✓ | ✓✓✓✓ | ✓✓ | ✓✓✓✓ | ✗✗✗✗ | ✓✓✓ | ✓✓✓✓ | ✓ | ✓✓✓✓ |

Table 8: Significance tests on the change in features between a pretrained GPT-2 model and a trained model. In order, these features are: Length (in characters), Brunet Index (BI), Total Dependency Distance (TDD), Total Dependencies (TD), Circuitousness, Punctuation Rate, Past, Pronoun Rate, Closed Class Word (CCW) Rate, Token Type (TTR) Ratio, Average Dependency Distance (ADD), Average Dependencies (AD). In the table, the number of checks or crosses indicates the level of p-value and the correctness of the direction of the trend. Note that ✓: $p < 0.05$, ✓✓: $p < 0.01$, ✓✓✓: $p < 0.001$, ✓✓✓✓: $p < 0.0001$.

verb rate, demonstrative rate, adjective rate, adposition rate, adverb rate, auxiliary rate, conjunction rate, determiner rate, interjection rate, numeral rate, particle rate, pronoun rate, proper noun rate, punctuation rate, subordinating conjunction rate, symbol rate, possessive rate), ratios of part of speech (e.g. noun-verb ratio, noun ratio, pronoun-noun ratio, closed-class word rate, open-class word rate), dependency information (total dependency distance, average dependency distance, total dependencies, average dependencies), content density, idea density, lexical diversity statistics (Honore statistic, Brunet index), type token ratio, average word length, proportion of inflected verbs, proportion of auxiliary verbs, proportion of gerund verbs, proportion of participles, proportion of mispelled words, amount of alliteration, passive voice, average number of syllables, proportion of jargon, proportion of MTCG verbs (Modal, Tentative, Certainty, Generalizing), rates of named-entity recognition (NER) tags (e.g. PERSON, DATE, CARDINAL, WORK OF ART, NORP, Certainty, GPE, ORG, LOC, PERCENT, MONEY, QUANTITY, TIME, PRODUCT, EVENT, LANGUAGE, FAC), and word embedding-based measures (i.e. speed, volume, circuitousness) (Toubia et al., 2021). The NER tags were obtained from the spaCy library (Honnibal and Montani, 2017) and many of the linguistic features are obtained from the blabla library (Shivkumar et al., 2020).

### C.2 Bayesian Model

We define the hierarchical Bayesian model as a binomial distribution around $p$. Note that text A always demonstrates the style more strongly than text B or equally to text B. We calculate $p$ as follows:

$$logit_p = \bar{p} + (\alpha[A_{id}] - \beta[B_{id}]) + \gamma[t] * (A_{ft} - B_{ft}) \quad (5)$$

where $\bar{p}$ is the intercept, $\alpha$ and $\beta$ are meant to capture any existing bias towards either text and $\gamma$ measures the correlation between the linguistic feature and the style for a specific topic $t$. We construct $\alpha$ and $\beta$ for all texts, hence why we index

Table 9: Percentage agreement with the linguistic feature correlations calculated using the hierarchical Bayesian model. Baseline models: GPT2, GPT-2 fine-tuned on UKPConvArg1 or the Cornell Movie Quotes corpus, GPT-2 with augmented data, and TitleStylist (Jin et al., 2020). Our models are trained on augmented data and a sample-dependent discriminator (SD) or sample-dependent supervised (SS) loss with parameter $\beta$. We show that our models are significantly better at learning stylistic features compared to our baselines.

| MODEL | PERSUASIVENESS | MEMORABILITY |
|---|---|---|
| GPT2 | 29.51 | 40.71 |
| GPT2-FT | 41.03 | 46.22 |
| GPT2-Aug | 44.01 | 51.26 |
| TStylist | 35.76 | 43.86 |
| SS-0.1 | 42.26 | 49.70 |
| SS-1.0 | 48.57 | 44.97 |
| SD-0.1 | 43.58 | 48.73 |
| SD-0.5 | 48.56 | **62.35** |
| SD-0.8 | 50.04 | 48.14 |
| SD-1.0 | **50.18** | 55.61 |

them with $A_{id}$ and $B_{id}$, respectively. Similarly, we construct $\gamma$ for each topic. $A_{ft}$ and $B_{ft}$ are the features of text A and B, respectively. $\alpha$, $\beta$, and $\gamma$ are all constructed similarly. Let's take $\alpha$ as an example:

$$\alpha = \bar{\alpha} + \alpha_v \alpha_\sigma \tag{6}$$

where $\bar{\alpha} \sim \mathcal{N}(0, 0.25)$ and $\alpha_\sigma$ is drawn from an exponential distribution with $\lambda = 1$. We construct a separate $\alpha_v$ for each unique $A_{id}$ where each $\alpha_v \sim \mathcal{N}(0, 1.0)$. These values are chosen because they help the MCMC sampling converge. $\beta$ and $\gamma$ follow the same construction except with different shapes for $\beta_v$ and $\gamma_v$. During the training of the hierarchical Bayesian model, we use 1000 warmup steps and generate an additional 1000 samples.

### C.3 Feature Agreement

To demonstrate feature agreements, we calculate a weighted average to quantify the results of Table 7 in the context of the full feature set, using the correlations obtained from the Bayesian model as weights. The results are shown in Table 9.