DRAG: Director-Generator Language Modelling Framework for Non-Parallel Author Stylized Rewriting

Anonymous EACL submission

Abstract

Author stylized rewriting is the task of rewriting an input text in a particular author's style. Recent works in this area have leveraged Transformer-based language models in a denoising autoencoder setup to generate author stylized text without relying on a parallel corpus of data. However, these approaches are limited by the lack of explicit control on target attributes and being entirely data-driven. In this paper, we propose a Director-Generator framework to rewrite content in target author's style, specifically focusing on certain target attributes. We show that our proposed framework works well even with limited-sized target author corpus. Our experiments on corpora consisting of relatively small-sized text authored by three distinct authors show significant improvements upon existing works to rewrite input texts to target author's style. Our quantitative and qualitative analyses further show that our model has better meaning retention and results in more fluent generations.

1 Introduction

With recent advances in language modelling techniques that have resulted in powerful language models (Radford et al., 2019; Devlin et al., 2018; Brown et al., 2020) along with an increased interest in stylized content generation (Hu et al., 2017; Shen et al., 2017; Subramanian et al., 2018; Fu et al., 2018; Niu and Bansal, 2018), large language models have been successfully tuned to achieve text stylization (Lample et al., 2018; Ziegler et al., 2019; Syed et al., 2020). Apart from *transferring* an input text to the target style, which has received recent interest from the community, *understanding* and *measuring* style have been persistently explored over the last few decades (Kessler et al., 1997; Garera and Yarowsky, 2009; Liu, 2012; Verma and Srinivasan, 2019). Lying at the intersection of style transfer enabled by advanced language models and

a deep understanding of style as a nuanced combination of several linguistic concepts, problems like *stylized generation* or *stylized rewriting* have gained further traction. A large body of work in style transfer focuses on binary aspects such as positive-negative sentiment (Li et al., 2018; Ziegler et al., 2019), formal-informal (Jain et al., 2019), and sometimes a mixture of these attributes (Subramanian et al., 2018). To fuel this interest in such binary stylization, some datasets comprising of text from the extreme ends of these spectrums have also emerged (e.g., positive-negative sentiments (Mathews et al., 2016), formal-informal (Rao and Tetreault, 2018)). As pointed by Syed et al. (2020), author stylized rewriting does not directly fit under any of these variants as writing style of an author is an amalgamation of several such attributes and needs to be modeled in a fine-grained manner.

Apart from the distinction along style dimensions, prior works can also be categorized as supervised (using parallel corpus (Jhamtani et al., 2017)) and unsupervised (Li et al., 2018; Syed et al., 2020; Niu and Bansal, 2018). In supervised frameworks, parallel data is used to tune sequence-to-sequence models for stylized rewriting. However, annotating such parallel corpus is a tedious effort and therefore, there is an increased interested in unsupervised style transfer; i.e., when there is no direct supervision or parallel data available for training the models. In this work, we focus on such an unsupervised setting.

Existing approaches on unsupervised author stylized rewriting rely on implicitly learning the target stylistic attributes from data and do not allow finer control on generation (Syed et al., 2020). While this is a good starting point for author-stylized rewriting, it is desirable to further improve the rewriting model on certain aspects without compromising on other attributes that the model has already optimized. An example would be to retain

the stylistic strengths while improving the content retention, or vice versa. To this end, we propose **Di**recting **a G**enerator framework (DRAG). Our quantitive and qualitive experiments show the viability of the proposed approach. Experiments further indicate that the framework's setup allows it to operate efficiently in low data setting and improves the performance over the baseline models. Our contributions can be summarized as - **(1)** We introduce a director-generator approach to *rewrite* an input text in a target author's style. **(2)** We propose linguistic alignment scores – both at local and global level and extend these to design *thresholds* for the generator and director. **(3)** We present experimental results on texts written by three authors from the Gutenberg corpus with very distinct writing styles, and show that our approach outperforms prior works across content retention and style alignment metrics. **(4)** We further identify and discuss shortcomings of our proposed approach, and present error analysis to aid future research in author stylized rewriting.

2 Related Work

With the rise of Transformer-based (Vaswani et al., 2017) **language models**, generative pretraining (Devlin et al., 2018; Radford et al., 2019; Brown et al., 2020) has advanced the field of NLP significantly. Fine-tuning such large language models on specific task has become very prevalent (Sun et al., 2019; Lee et al., 2020; Lample and Conneau, 2019; Raffel et al., 2019; Liu et al., 2019). Pretraining infuses the *generic* language knowledge into the language model helping it understand the specific tasks with relatively much less supervision. In fact, recent approaches (Radford et al., 2019; Brown et al., 2020) show that often, even such small supervisions are not required and a simple instruction can be used to solve specific tasks by utilizing the capabilities of such large language models trained on very large datasets.

Pretraining of such models usually involves optimizing them on **Masked Language Modelling (MLM)** (Devlin et al., 2018), **Causal Language Modelling (CLM)** (Radford et al., 2019) or other similar (Clark et al., 2020) objectives. While CLM is the task of autoregressively predicting the next word given the previous words/context, MLM is the task of recovering masked tokens from a given input. While these approaches mostly train only an encoder or a decoder framework, Lample and Conneau (2019) explored initializing encoder-decoder

frameworks using the pretrained encoders for cross-lingual translation. Such a technique with appropriate modification has been shown to be successful in incorporating stylistic aspects of the language as well (Conneau and Lample, 2019; Syed et al., 2020). All these works utilize the task of minimizing the *denoising auto-encoder loss* for inducing style in the language models in a reconstruction framework. For our explorations, we leverage these works to initialize our DRAG framework.

There is an increased interest in **stylistic generation** or rewriting of content. Most of the approaches define dimensions like formality-informality (Shen et al., 2017; Ficler and Goldberg, 2017; Jain et al., 2019; Sun et al., 2019) and achieve the alignment along these dimensions. While some of these approaches rely on parallel corpus (Ficler and Goldberg, 2017; Jhamtani et al., 2017), many of the approaches focus on unsupervised framework (Li et al., 2018; Shen et al., 2017; Jain et al., 2019), where the model *preserves* the input content in the output while *biasing* the generations towards the target style. While some approaches utilize simple *editing* to achieve the style along particular dimensions (Li et al., 2018), others focus on achieving this through discriminators (Fu et al., 2018) or scorers (Jain et al., 2019). As mentioned before, since author style is an amalgamation of several such attributes, it requires much more than a discriminator or singular dimension tuning to achieve stylization.

Due to the difficulty associated with author style understanding and fine-grained nature of those styles even if understood, the problem of **author stylized rewriting** has not been explored enough. While, Jhamtani et al. (2017) try to solve this problem for a specific author (i.e. Shakespeare), their approach is contingent on the availability of parallel corpus. Since preparing parallel corpus is a tedious and intractable process, especially while dealing with multiple authors and multiple combinations of input and output styles, it is essential to focus on unsupervised solutions. Most recently, Syed et al. (2020) leverage the capabilities of the large language models to solve this problem in an unsupervised manner. However, their approach does not explicitly encourage stylization along specific dimensions and rely solely on the data, thus providing no explicit control. In this paper, we leverage their framework for stylized rewriting and introduce a *director-generator* frame-

work to further guide the model towards certain target attributes.

3 Author Style

There has been significant work on understanding binary stylization along dimensions like formal-informal, positive-negative sentiment (Rao and Tetreault, 2018; Kessler et al., 1997; Pavlick and Tetreault, 2016; Collins-Thompson and Callan, 2005; Hovy, 1990; Inkpen and Hirst, 2006; Kantrowitz, 2003), however, there is limited work on understanding an author's writing style (McCarthy et al., 2006; Forgeard, 2008; Verma and Srinivasan, 2019). While style can be a mixture of several factors including, but not limited to, lexical preferences, syntactic/sentential choices, discourse structure, narrative style, tone, we follow Syed et al. (2020) and consider author's style at three levels:

Surface style is estimated using the frequencies of different surface elements such as the number of comma, semicolon, colon, question mark, exclamation mark, and hyphen per paragraph, from a given author's text. We, thus, quantify the surface-style elements into a 6-dimensional vector.

Lexical style of an author is reflected in the author's choice of words. To describe the same concept, different authors may use different words. For instance, Rudyard Kipling, known for his classics in children's literature, had a tendency to use more concrete words (e.g., gongs, rockets, torch) while Abraham Lincoln, being a political writer, used more abstract words (e.g., freedom, patriotism). We enumerate lexical style categories as subjective, objective, literary, colloquial, abstract and concrete (Brooke and Hirst, 2013). We use lexicons for each of these categories (Brooke and Hirst, 2013), and define *lexical style alignment* of each word in the vocabulary to a given style category as the average and normalized point-wise mutual information (PMI) between that word and the seed words in the lexicon for that style category. The lexical style alignment for each word is thus a 6-dimensional vector. We use the EmoBank corpus (Buechel and Hahn, 2017) to compute the co-occurrence statistics for PMI computations. The inclination of a word towards a style category is positive, if its normalized PMI score is positive with respect to the given category. The inclination of an author towards a style category is then estimated by the fraction of words in their text that have positive inclination towards the category.

Syntactic style of an author is indicated by the nature of sentences used and we estimate the distribution of different types of sentences in an author's text. Sentence types may range from complex, as seen in philosophical writings, to simple, as observed in children's story books. We use five categories of sentence styles: (i) simple, (ii) compound, (iii) complex, (iv) complex-compound sentences, and (v) others (Feng et al., 2012; Verma and Srinivasan, 2019; Syed et al., 2020). Sentences are categorized into one of these types using the algorithm proposed by (Feng et al., 2012). The resulting 5-dimensional probability distribution vector is used as the estimation of syntactic style. These vectors are estimated at corpus-level, unlike those for lexical and surface style which are computed at paragraph-level.

4 DRAG: *Directing a Generator for Stylized Rewriting*

Our proposed framework, DRAG, that aims to rewrite a given piece of text with a specific target author's style consists of three main stages:
(1) Pretraining a language model to infuse general linguistic knowledge into the model
(2) Adapting the pretrained language model towards the target author's writing style by further pretraining it on text written by this author (Syed et al., 2020), and
(3) Using a director-generator framework (as discussed later) to fine-tune such *biased* language model to improve its style transfer capabilities even further while fixing content preservation issues. It is worth noting that we do not rely on the availability of parallel data for any of our experiments.

4.1 Pretraining Language Model

In order to infuse general linguistic knowledge into a language model, we leverage Tranformer-based pretrained language models (Devlin et al., 2018; Radford et al., 2019; Brown et al., 2020) due to their recent success in text processing tasks (Vaswani et al., 2017; Devlin et al., 2018; Brown et al., 2020). Similar to Conneau and Lample (2019), we first train a Transformer-based encoder on the Masked Language Modelling (MLM) task with 15% of the tokens masked (Devlin et al., 2018) on a *generic* text corpus. We initialize an encoder-decoder framework, as shown in Figure 1, with this language model.

4.2 Adapting LM for Rewriting

To adapt the pretrained LM for author stylized rewriting, Syed et al. (2020) initialize an encoder-
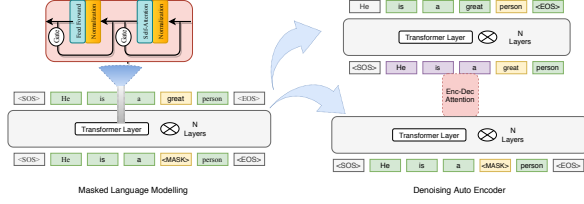
Figure 1: Language Model Pretraining using Masked Language Modelling followed by encoder-decoder initialization using pretrained models. This process still leaves the encoder-decoder attention parameters unitialized which can be initiliazed using the Denoising Auto Encoder training as depicted in the figure.

decoder framework with the pretrained LM, as shown in Figure 1. This is followed by optimizing it on denoising auto-encoder (DAE) loss (Lample et al., 2018; Lample and Conneau, 2019) *only* over target author's corpus. Syed et al. (2020) use the DAE loss to infuse an author's linguistic style into the reconstruction model; we refer to this framework as STYLELM. The fine-tuning using the DAE loss on a target author's corpus encourages recovering actual paragraphs from their noisy version (Lample and Conneau, 2019). For a paragraph g in corpus G and its noisy version $C(g)$ ($C(.)$ being the noise function), DAE loss is given by,

$$DAE(\theta_e, \theta_{ed}, \theta_d) = -\frac{1}{|G|} * \sum_{g \sim G} \log P(g/C(g); \theta_e, \theta_{ed}, \theta_d)$$
(1)

where P is the probability of reconstruction for a given encoder parameters θ_e, decoder parameters θ_d, and encoder-decoder attention parameters θ_{ed}. Please note that θ_{ed} does not refer to any additional layer but the parameters which are present in transformers and are responsible for encoder-decoder attention. In our setup, $C(.)$ function introduces two noises: (a) random dropping of words with 10% probability, and (b) word masking by replacing it with [MASK] token with 10% probability.

Given a noisy input, the encoder fills the [MASK] tokens with suitable replacements (based on the knowledge from its MLM pretraining), thus creating a pseudo *generic* input for the decoder, the target sequence for which is aligned to the target author's style. However, we identify and verify experimentally two issues with this approach:

(1) It requires a large target author corpora to achieve meaningful content preservation capability. This is evident by its very low content preservation scores (as discussed in Section 5.2) when trained on authors with relatively smaller corpora. Even with large corpora, the model still suffers from ex-posure bias to texts written only by the target author leading to spurious outputs for *unseen* inputs.

(2) The masking results in a significant emphasis on lexical style aspects, with lesser focus on the surface and syntactic preferences. Since the model is completely data-driven, there is no way to explicitly add emphasis on additional style aspects.

One of the primary reasons behind (1) is the lack of explicit initialization of encoder-decoder attention parameters in STYLELM resulting in a random initialization. The model therefore needs a large corpus of author data to stabilize these parameters. To fix this, we propose to train the entire encoder-decoder language model using the DAE loss over the same *generic* corpus used for pre-training. The resulting model will be in the *generic* language space (English, in our case), and henceforth referred to as VANILLALM. We, further, finetune VANILLALM in the author corpus on the DAE loss to arrive at an improved version of STYLELM which we call ISTYLELM. This offers better encoder-decoder attention initialization, and also removes the exposure bias of STYLELM, thus resulting in a more resilient and stable model with improved content preservation abilities (as demonstrated in Section 5.2).

However, at this point, we note that ISTYLELM still fails to address (2), and its content preservation ability is also sub-optimal as the target author's style aspects which are infused at the later stage of training override some of the general linguistic knowledge. To further improve on ISTYLELM, we introduce a Director-Generator component to our training framework in the next section.

4.3 Director-Generator Finetuning

For the Director-Generator finetuning, we find inspiration in the standard RL strategies (Rennie et al., 2017; Ranzato et al., 2015) where the *nearby* space is *explored* and certain actions are rewarded higher than others, consequently getting encouraged in the future. We, however, find direct rewarding unstable for our problem. Hence, we *generate* potential *directives* during exploration and *accept* or *reject* them on the basis of thresholds. A *directive*, in our context, is as an output paragraph generated from an input by a director model which is fixed and has been initialized using ISTYLELM. Specifically, we create two copies of the ISTYLELM as the *Director* and the *Generator*. As the names indicate, for each input, the direc-
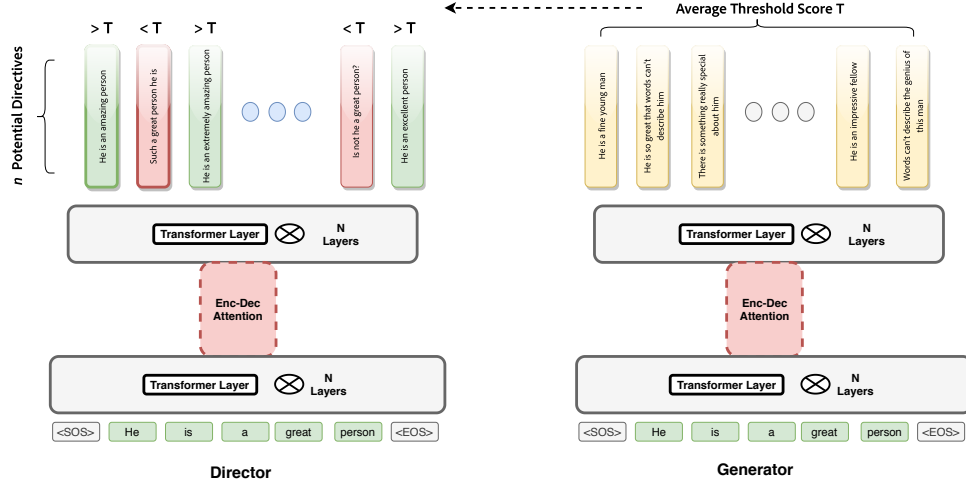
Figure 2: Both the director as well as generator, intiliazed using ISTYLELM, work together to improve the final outputs. While director remains in the space of author style generating and exploring potential directives , generator keeps changing its threshold as it gets improved on its content & style capabilities. The directives above the average threshold for same example are accepted while rest of them are rejected.

tor proposes n potential directives or paragraphs, while the generator generates n *thresholding* outputs (paragraphs) as shown in Figure 2. We generate the potential directives using nucleus sampling (Holtzman et al., 2019) with a softmax temperature of 1.2, while the thresholding outputs are generated using a softmax temperature of 0.8 (same value is used at inference time as well). We score the director and generator outputs on various content and style attributes. For content preservation, we use the BLEU score between input and output as the content score. For lexical style, the mean squared error is calculated between the 6-dimensional lexical alignment vector of the directives/generator outputs (calculated as the averaged sum of alignments of words in the proposal) and average lexical alignments of paragraphs for the target author corpus. Similarly, mean squared error for surface style is also calculated. The scores L and S for lexical and surface styles, respectively, are then calculated as reciprocal of means squared errors (with ϵ added in denominator to avoid zero-division). For syntactical choices, since we wish to achieve the probability distribution of different types of sentences at the corpus-level, we calculate the score for syntactic style as, $SX = \frac{sum(P_p \circ P_t)}{sum(P_p)}$ where P_p denotes the frequency distribution of different types of sentences in a directive/generator's output, P_t the probability distribution of different types of sentences in target author corpus, and $\circ$ denotes the Hadamard product. All three scores are summed to calculate the style score for the directives (and the generator outputs).

The ISTYLELM model already captures certain stylistic aspects of the target author. We want our model to leverage this understanding and improve on aspects where ISTYLELM does not perform well. To capture this, we compute the content and style scores of all the potential directives and generator outputs, and retain *only* those directives which have **both** the content and style scores better than the average of the generators' outputs' scores. The *accepted* directives become *real* directives for the generator and are used to train it using the *teacher-forcing* cross-entropy loss. Note again that the director remains frozen with ISTYLELM during the entire training process. In the case of multiple potential directives being better than the generators' outputs' average, the cross-entropy loss for each directive is weighted by its marginal difference from generator's average score on the style dimension; i.e., if the style score for a directive is D_s and average outputs' style score from generator is G_s, its weight during the cross-entropy training is $D_s - G_s$. This objective is similar to the one used in SCST (Rennie et al., 2017) but only *accepted* directives are encouraged and nothing is explicitly discouraged.

In order to stabilize the Director-Generator fine-tuning framework, we use (a) **fixed director**, and (b) **moving generator**. Contrary to the natural expectation of exploring better directives with training of the director, the fixed or frozen director prevents catastrophic degradation in case the training biases the model towards specific choices which further train the model. It is a known phenomenon

in RL frameworks that the model quickly learns to bias towards specific choices that are more rewarding. Specifically, we observe that training the director as well leads to overfitting to the limited stylistic choices, thus resulting in exploration of sub-optimal potential directives that seldom cross the required thresholds, especially the content preservation ones. With a moving (i.e. trained at each step) generator, its outputs scores accounts for the *current* state of the model against a fixed stable director, and hence only those directives get accepted which are better than the *current* capabilities (thresholds) of the generator. With fixed generator, directives which would have been worse than *current* capabilities of model but better than the capabilities of fixed generator would also get accepted, thus training the model in the opposite direction. The Director-Generator finetuned IStyleLM yields our proposed DRAG framework. At the inference time, we drop the director and use Generator as our final rewriting model.

5 Experiments

We use a transformer encoder[1] with 512 hidden units, 16 heads, a dropout rate of 0.1, and learned positional embeddings during our MLM training. The model is trained using Adam Optimizer with a learning rate of 10^{-4}. The batch size used is 32 with a stream of 256 tokens, and the whole setup is trained until the validation performance (perplexity scores) shows no further improvement. The Transformers used in encoder-decoder setup also have the same parameters, and are initialized using the above encoder before training on further objectives. During DAE loss training, we use the same hyperparameters used in (Conneau and Lample, 2019; Syed et al., 2020), and set p_{drop} and p_{blank} to 0.1. During director-generator training, we use n as 8 and ϵ as 0.05. The learning rate used in this case is 10^{-5}. In all the models, we use Byte Pair Encoding (Sennrich et al., 2015) with $80k$ codes learnt over the entire *generic* corpus.

5.1 Dataset

We use the 2,857 books written by 142 authors in the Gutenberg corpus (Lahiri, 2014), as used in (Syed et al., 2020), along with the Wikipedia articles, to form a corpus of about 4.6M passages. We refer to this corpus as *generic* during all our experiments, since it infuses only *generic* linguis-

tic knowledge into the models. While MLM and VanillaLM are trained on the *generic* corpus, we select three authors with the most distinct writing styles, namely Albert Einstein, Michael Faraday, and John Stuart Mill, as measured by comparing their lexical alignments with the average lexical alignment of the Gutenberg corpus, as the target authors for author-specific style rewriting. Note that the choice of the authors is made purely on statistical basis with these three authors having maximum lexical style difference on their style vectors as described earlier when compared with the lexical style of entire *generic* corpus. For evaluation, we use the Opinosis corpus (Ganesan et al., 2010) as well as mixed author Gutenberg subset (with five passages from all the authors except the target author), which we refer to as *Generic (Test)*.

5.2 Quantitative Evaluation

Table 1 shows the results averaged over the three selected authors. The experiments are conducted on Opinosis and *Generic (Test)* datasets, using the following four models.

- **VanillaLM** is initialized using MLM-trained encoders and decoders and fine-tuned on the generic corpus using DAE loss.
- **StyleLM**, proposed by Syed et al. (2020),[2] is also initialized using MLM-trained encoders and decoders, but fine-tuned *only* on the target author corpus (instead of the generic corpus).
- **IStyleLM**, an improved and stronger baseline compared to StyleLM, is initialized with VanillaLM and then fine-tuned on the target author corpus.
- **DRAG** is our proposed model. We use IStyleLM to initialize both director and generator as described above, and then fine-tune them using inputs from *generic* corpus.

While the *Generic(Test)* corpus is predominantly literary due to the nature of the source, Opinosis covers everyday language. As shown in Table 1, StyleLM improves on the style alignment scores, but at a great cost of content preservation when the target author corpus is small. This is possibly due to the random initialization of encoder-decoder attention parameters in the DAE training over target corpus, as reflected in the superior performance of IStyleLM. We also note that while

---

[1] As proposed by Parisotto et al. and shown in Fig. 1

[2] Note that the StyleLM code is not publicly available. Results shown in the table are using our own implementation.

Dataset	Model	Content Preservation (↑)				Author Style (↓)		
		ROUGE-1	ROUGE-2	ROUGE-L	BLEU	Lexical(RMSE)	Surface(RMSE)	Syntactic(JSD)
Opinosis	Vanilla LM	75.23	56.12	74.28	59.46	0.232	2.74	0.132
	StyleLM (Syed et al., 2020)	32.61	14.51	31.69	14.82	**0.174**	2.56	0.121
	iStyleLM	49.28	29.16	49.86	28.16	0.178	2.61	0.122
	DRAG	**57.23**	**36.12**	**56.98**	**37.53**	0.179	**2.48**	**0.109**
Generic	Vanilla LM	72.34	54.65	71.93	56.46	0.218	2.48	0.120
	StyleLM (Syed et al., 2020)	30.31	11.58	29.77	12.91	**0.163**	2.29	0.114
	iStyleLM	45.12	26.08	44.36	24.16	0.171	2.33	0.121
	DRAG	**52.39**	**30.66**	**51.98**	**33.28**	0.174	**2.21**	**0.097**

Table 1: ↑ indicates higher scores are better while ↓ indicates the opposite. Apart from lexical alignment where StyleLM performs marginally better, DRAG outperforms prior approaches. Vanilla LM performs best at content preservation but lacks any stylization

Input	Albert Einstein	Michael Faraday	John Stuart Mill
The accuracy at this point is very good	The experimental definitions developed is very *clearly*	The point is very wonderful	The physical for this **,** is very very pretty .
The estimated time to arrival does not seem to calculate the travelling time accurately	The estimated time relative to the leading existence does not seem likely to calculate the travelling time *exactly*	The discovery of ascertaining time **;** indeed **,** do not not show accuracy to the time to *angles*	The total time is to infer that arrival is not verified but often clearly a **,** accurately .

Table 2: Qualitative Outputs For Three different authors for same inputs

Input	StyleLM(Syed et al., 2020)	DRAG (Ours)
but after that it is very **easy** and quite **accurate** to use.	But for all that it is very After question about this and quite *measured* with consideration.	But after all it is very accurate and quite illustrious to the use of *events*.
Leather seats are very comfortable.	come on very very should we have any replaced.	This moving *hypothetical* seats are very comfortable.
I am not real fond of the electric seat and I find it is not as comfortable as my F150 pickup on trips	I am not real and use of the electric *position* and I find that it is not as well may's for the very hardly small have we led train	I am not real fond of the electric seat **,** and I find it is not as comfortable as *my physical relative on railway investigations.* .

Table 3: Comparison between STYLELM and DRAG for Albert Einstein

the approach proposed in STYLELM (Syed et al., 2020) improves lexical scores significantly, it fails to bring the same level of improvement in surface and syntactic alignments, perhaps due to the due to rare chances of less frequent punctuation symbols getting masked during DAE training, even more so when the target author corpus is not large enough to cover all possible masks. Similar reasoning explains the syntactic alignment issues, The DRAG approach, however, improves on both surface and syntactic alignment along with content preservation scores even though it comes at the marginal cost of lexical alignment. Please note that the purpose of Vanilla LM is to provide an estimation of *upper limit* on the content preservation scores and is not to be treated as a baseline due to the simple objective of its task (just copying the input tokens).

5.3 Qualitative Comparisons

We also qualitatively show some comparisons for different authors and different models. In Table 2, we show the outputs of DRAG for same input and different target authors. Evidently, our model produces changes both at the lexical as well as surface

levels. Word 'good' in the first input is replaced by words like 'clearly', 'wonderful' and 'pretty' depending on the author. Some words do not replace any word but still get added to change the syntactical structure of the sentences. For example - appearance of word 'relative' starts comparison to the 'leading existence' making it a bit complex. Sometimes surface level changes like appearance of ';' also change the complexity of sentences.

We also show the comparison between STYLELM and our proposed DRAG outputs for same inputs when the target author is Albert Einstein as shown in Table 3. Evidently, while both models try to achieve the stylistic alignment, STYLELM ends up distorting the input sentence too much resulting in poor content preservation properties. Words like 'measured', 'hypothetical', and 'physical' relative reflect the *objective* approach used in Albert Einstein's writings.

6 Discussions and Limitations

While the language generation advancements are happening at a very high pace, the notion of style and the ability of models to rewrite same content in different styles is still far from being solved.

One of the most important observation as made by (Lample et al., 2018) is that it is very difficult to separate content from style. In fact, previous approaches which worked on the principle of *disentangling* style from content were not found to disentangle the style so much after all (Lample et al., 2018). The notion of style is still very far from being defined and concretized. While some psycholinguistic concepts can be defined to some extent (formality, sentiment, etc.), defining it at the level of author's style is very difficult due to manifestation of style at different levels as enumerated by (Verma and Srinivasan, 2019). Despite, such enumeration at various levels, it is far from *exhaustive* and therefore our approach still requires more granular understanding of style to closely emulate target author's style.

Our evaluation uses automatic metrics for style due to the difficulty associated in conducting human evaluation in author attribution tasks(Syed et al., 2020). The skill needed to identify author's style is very intense thus making the human evaluation very costly. A more granular and detailed study on understanding how humans interpret an author's style is required to design a proper feedback mechanism. This is, however, outside the scope of this work.

6.1 What Did not work

In this section, we discuss some of our explorations which did not work as expected to aid future research in author stylization. We experimented with various **reinforcement learning setups** as it was a more natural choice once we had scoring engines for rewards. Using the VANILLALM as a policy and we explored Self Critical Sequence Training (SCST) (Rennie et al., 2017; Ranzato et al., 2015) and Proximal Policy Optimization(Schulman et al., 2017). However, all the setups were unstable in various ways for our problem. Note that our experiments and observations here are limited to the problem of author stylized rewriting only. SCST or self-critical sequence training is aimed at bringing the advantages of reinforcement learning setups for sequence level problems. A model (or policy) generates/explores outputs (or episodes) using multinomial sampling and greedy sampling. If the greedily sampled episode reward is r_b and non-greedily sampled episode reward is r - the whole setup is trained using REINFORCE(Sutton and Barto, 2018) with r as the actual reward and r_b as baseline reward.

We found this to limit exploration considering our problem is relatively much harder than previous metrics on which SCST has been successful due to our target metric being of exact value. It therefore resulted in no improvement in either style or content scores. We, therefore, shifted to its modified version to encourage exploration, where we generated multiple episodes for each input and averaged their scores to use that as baseline reward r_b and trained the setup on all generated episodes using REINFORCE (Sutton and Barto, 2018). We found this approach to be effective at style incorporation but not generalizable at all. The model learnt to repeat certain patterns with poor content preservation abilities. We, tried, to balance it with occasional denoising autoencoder loss training but that only *delayed* the overfitting and not solve it. We also attempted Proximal Policy Optimization in a setup same as (Sun et al., 2019) but it resulted in even worse outputs due to critic's failure to approximate complex value functions for our objectives.

As discussed already, we only accept those directives which have scores above threshold. We also tried a variant of it which had even those directives which do not score above threshold. We scored them negatively thereby resulting in a bit similar framework like SCST but within some steps, we found more negative scores than positive due to bad content preservation pushing the model away from a bad state towards some undefined state resulting in spurious and inconsistent outputs.

7 Conclusion and Future Work

In this work, we addressed the shortcomings of the prior approaches for the task of author stylized rewriting and overcame them through DRAG: a Director-Generator approach. We showed the effectiveness of our proposed approach for stylized rewriting on three different authors from the Guteneberg Corpus. Furthermore, we discussed the limitations of our approach and some of the failure cases to aid future research. While our DRAG approach is able to stabilize the training while improving content preservation abilities of the model, a standard reinforcement learning approach, when stabilized, has the potential to improve these scores to much more improved level. Improved understanding of author style while keeping a human in the loop and stabilizing RL with transformers models are subjects of future research.

References

Julian Brooke and Graeme Hirst. 2013. A multi-dimensional bayesian approach to lexical style. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 673–679.

Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*.

Sven Buechel and Udo Hahn. 2017. EmoBank: Studying the impact of annotation perspective and representation format on dimensional emotion analysis. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 578–585, Valencia, Spain. Association for Computational Linguistics.

Kevin Clark, Minh-Thang Luong, Quoc V Le, and Christopher D Manning. 2020. Electra: Pre-training text encoders as discriminators rather than generators. *arXiv preprint arXiv:2003.10555*.

Kevyn Collins-Thompson and Jamie Callan. 2005. Predicting reading difficulty with statistical language models. *Journal of the American Society for Information Science and Technology*, 56(13):1448–1462.

Alexis Conneau and Guillaume Lample. 2019. Cross-lingual language model pretraining. In *Advances in Neural Information Processing Systems*, pages 7059–7069.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Song Feng, Ritwik Banerjee, and Yejin Choi. 2012. Characterizing stylistic elements in syntactic structure. In *Proceedings of the 2012 joint conference on empirical methods in natural language processing and computational natural language learning*, pages 1522–1533.

Jessica Ficler and Yoav Goldberg. 2017. Controlling linguistic style aspects in neural language generation. *arXiv preprint arXiv:1707.02633*.

Marie Forgeard. 2008. Linguistic styles of eminent writers suffering from unipolar and bipolar mood disorder. *Creativity Research Journal*, 20(1):81–92.

Zhenxin Fu, Xiaoye Tan, Nanyun Peng, Dongyan Zhao, and Rui Yan. 2018. Style transfer in text: Exploration and evaluation. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Kavita Ganesan, ChengXiang Zhai, and Jiawei Han. 2010. Opinosis: A graph based approach to abstractive summarization of highly redundant opinions.

Nikesh Garera and David Yarowsky. 2009. Modeling latent biographic attributes in conversational genres. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 710–718.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2019. The curious case of neural text degeneration. *arXiv preprint arXiv:1904.09751*.

Eduard H Hovy. 1990. Pragmatics and natural language generation. *Artificial Intelligence*, 43(2):153–197.

Zhiting Hu, Zichao Yang, Xiaodan Liang, Ruslan Salakhutdinov, and Eric P Xing. 2017. Toward controlled generation of text. *arXiv preprint arXiv:1703.00955*.

Diana Inkpen and Graeme Hirst. 2006. Building and using a lexical knowledge base of near-synonym differences. *Computational linguistics*, 32(2):223–262.

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. *arXiv preprint arXiv:1707.01161*.

Mark Kantrowitz. 2003. Method and apparatus for analyzing affect and emotion in text. US Patent 6,622,140.

Brett Kessler, Geoffrey Nunberg, and Hinrich Schütze. 1997. Automatic detection of text genre. *arXiv preprint cmp-lg/9707002*.

Shibamouli Lahiri. 2014. Complexity of Word Collocation Networks: A Preliminary Structural Analysis. In *Proceedings of the Student Research Workshop at the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 96–105, Gothenburg, Sweden. Association for Computational Linguistics.

Guillaume Lample and Alexis Conneau. 2019. Cross-lingual language model pretraining. *arXiv preprint arXiv:1901.07291*.

Guillaume Lample, Sandeep Subramanian, Eric Smith, Ludovic Denoyer, Marc'Aurelio Ranzato, and Y-Lan Boureau. 2018. Multiple-attribute text rewriting. In *International Conference on Learning Representations*.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2020. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

9

Juncen Li, Robin Jia, He He, and Percy Liang. 2018. Delete, retrieve, generate: A simple approach to sentiment and style transfer. *arXiv preprint arXiv:1804.06437*.

Bing Liu. 2012. Sentiment analysis and opinion mining. *Synthesis lectures on human language technologies*, 5(1):1–167.

Nelson F Liu, Matt Gardner, Yonatan Belinkov, Matthew E Peters, and Noah A Smith. 2019. Linguistic knowledge and transferability of contextual representations. *arXiv preprint arXiv:1903.08855*.

Alexander Patrick Mathews, Lexing Xie, and Xuming He. 2016. Senticap: Generating image descriptions with sentiments. In *Thirtieth AAAI conference on artificial intelligence*.

Philip M McCarthy, Gwyneth A Lewis, David F Dufty, and Danielle S McNamara. 2006. Analyzing writing styles with coh-metrix. In *FLAIRS Conference*, pages 764–769.

Tong Niu and Mohit Bansal. 2018. Polite dialogue generation without parallel data. *Transactions of the Association for Computational Linguistics*, 6:373–389.

Emilio Parisotto, H Francis Song, Jack W Rae, Razvan Pascanu, Caglar Gulcehre, Siddhant M Jayakumar, Max Jaderberg, Raphael Lopez Kaufman, Aidan Clark, Seb Noury, et al. 2019. Stabilizing transformers for reinforcement learning. *arXiv preprint arXiv:1910.06764*.

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

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

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*.

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

Sudha Rao and Joel Tetreault. 2018. Dear sir or madam, may i introduce the gyafc dataset: Corpus, benchmarks and metrics for formality style transfer. *arXiv preprint arXiv:1803.06535*.

Steven J Rennie, Etienne Marcheret, Youssef Mroueh, Jerret Ross, and Vaibhava Goel. 2017. Self-critical sequence training for image captioning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7008–7024.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2015. Neural machine translation of rare words with subword units. *arXiv preprint arXiv:1508.07909*.

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.

Sandeep Subramanian, Guillaume Lample, Eric Michael Smith, Ludovic Denoyer, Marc'Aurelio Ranzato, and Y-Lan Boureau. 2018. Multiple-attribute text style transfer. *arXiv preprint arXiv:1811.00552*.

Chi Sun, Xipeng Qiu, Yige Xu, and Xuanjing Huang. 2019. How to fine-tune bert for text classification? In *China National Conference on Chinese Computational Linguistics*, pages 194–206. Springer.

Richard S Sutton and Andrew G Barto. 2018. *Reinforcement learning: An introduction*. MIT press.

Bakhtiyar Syed, Gaurav Verma, Balaji Vasan Srinivasan, Anandhavelu Natarajan, and Vasudeva Varma. 2020. Adapting language models for non-parallel author-stylized rewriting. In *AAAI*, pages 9008–9015.

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 *Advances in neural information processing systems*, pages 5998–6008.

Gaurav Verma and Balaji Vasan Srinivasan. 2019. A lexical, syntactic, and semantic perspective for understanding style in text. *arXiv preprint arXiv:1909.08349*.

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*.