Transforming Delete, Retrieve, Generate Approach for Controlled Text Style Transfer

Akhilesh Sudhakar, Bhargav Upadhyay, Arjun Maheswaran
Agara
{akhilesh, bhargav, arjun}@agaralabs.com

Abstract

Text style transfer is the task of transferring the style of text having certain stylistic attributes, while preserving non-stylistic or content information. In this work we introduce the Generative Style Transformer (GST) - a new approach to rewriting sentences to a target style in the absence of parallel style corpora. GST leverages the power of both, large unsupervised pre-trained language models as well as the Transformer. GST is a part of a larger 'Delete Retrieve Generate' framework, in which we also propose a novel method of deleting style attributes from the source sentence by exploiting the inner workings of the Transformer. Our models outperform state-of-art systems across 5 datasets on sentiment, gender and political slant transfer. We also propose the use of the GLEU metric as an automatic metric of evaluation of style transfer, which we found to compare better with human ratings than the predominantly used BLEU score.

1 Introduction

Text style transfer is an important Natural Language Generation (NLG) task, and has wide-ranging applications from adapting conversational style in dialogue agents (Zhou et al., 2017), obfuscating personal attributes (such as gender) to prevent privacy intrusion (Reddy and Knight, 2016), altering texts to be more formal or informal (Rao and Tetreault, 2018), to generating poetry (Yang et al., 2018). The main challenge faced in building style transfer systems is the lack of parallel corpora between sentences of a particular style and sentences of another, such that sentences in a pair differ only in style and not content (non-stylistic part of the sentence). This has given rise to methods that circumvent the need for such parallel corpora.

Previous approaches using non-parallel corpora, that employ learned latent representations to disentangle style and content from sentences, are typically adversarially trained (Hu et al., 2017; Shen et al., 2017; Fu et al., 2018). However these models a) are hard to train and take long to converge, b) need to be re-trained from scratch to change the trade-off between content retention and style transfer c) suffer from sparsity of latent disentangled representations, d) produce sentences of bad quality (according to human ratings) and e) do not offer fine-grained control over target style attributes.

Li et al. (2018) find that style attributes are more often than not, localized to a small subset of words of a sentence. Building on this inductive bias, they model style transfer in a "Delete Retrieve Generate" framework (hereby referred to as DRG) which aims to 1) delete only the set of attribute words from a sentence to give the content, 2) retrieve attribute words from the target style corpus, and 3) use a neural editor (an encoder-decoder LSTM) to generate the final sentence from the content and retrieved attributes.

While DRG as a framework leads to output sentences that are better in quality than previous approaches, their individual Delete and Generate methods are susceptible to: a) removing core content words which would preserve crucial context, b) failing to remove source style attributes that should be replaced with target style attributes, c) the LSTM-based encoder-decoder model not being robust to errors made by the Delete and Retrieve models, d) generating sentences that are not fluent, by abruptly forcing retrieved attributes into the source sentence and e) failing on longer input sentences.

In this work, we propose a novel approach to rewrite sentences into a target style, that leverages the power of both a) transfer learning by using an unsupervised language model trained on a large corpus of unlabeled text, as well as b) the Trans-

Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, pages 3269–3279,
Hong Kong, China, November 3–7, 2019.

former (Vaswani et al., 2017). We refer to our Transformer as the Generative Style Transformer **(GST)**. We use the DRG framework proposed by Li et al. (2018) but we overcome the shortcomings of their a) Delete mechanism, by using the attention weights of another Transformer that we refer to as the Delete Transformer **(DT)**, and b) Generate mechanism by using **GST**, which does away with the need for (and consequent shortfalls of) a sequence-to-sequence encoder-decoder architecture using LSTMs.

We outperform the current state-of-art systems on transfer of a) sentiment[1], b) gender and c) political slant. Our approach is advantageous in that it is simple, controllable and exploits the important inductive bias described, while at the same time it leverages the power of Transformers in novel ways.

All code, data and results for this work can be found in our Github repository [2].

2 Our Approach

Given a dataset $D = \{(x_1, s_1), ..., (x_m, s_m)\}$ where x_i is a sentence and $s_i \in S$ is a specific style, our goal is to learn a conditional distribution $P(y|x, s^{tgt})$ such that $Style(y) = s^{tgt}$, where style is determined by an oracle that can accurately determine the style of a given sentence. For instance, for the sentiment transfer task, $S = \{$'Positive', 'Negative'$\}$. Using the DRG framework, we model our task in 3 steps:
(1) A **Delete** model which learns $P(c, a|x)$ such that c and a are non-stylistic and stylistic components of x respectively, $Style(c) \notin S$ (i.e., c does not have any particular style) and x can be completely reconstructed from c and a, **(2)** A **Retrieve** model which retrieves a set of (optional) target attributes a^{tgt} from $D_{s^{tgt}}$, the corpus of sentences of target style, and **(3)** A **Generate** model in two flavors: a) one which learns to generate a sentence in the target distribution $P(y|c, s^{tgt})$ and b) another which learns to generate a sentence in the target distribution $P(y|c, a^{tgt})$, both such that $Style(y) = s^{tgt}$. We now elaborate on each of these components individually.

[1]We use style in a broad socio-linguistic sense that encompasses sentiment too, for the purpose of this work

[2]https://github.com/agaralabs/transformer-drg-style-transfer

2.1 Delete

For an input sentence "The restaurant was *big* and *spacious*", in the case of a style transfer task from positive to negative sentiment, the Delete model should be capable of deleting the style attributes *big* and *spacious*.

Our approach to attribute deletion is based on 'input reduction' (Feng et al., 2018), based on the observation that certain words and phrases significantly contribute to the style of a sentence. For a sentence x of style s_j having a set of attributes a, a style classifier will be confused about its style if the attributes in a are removed from x. We describe a mechanism to assign an *importance* score to each token in x, which is reflective of its contribution to style. These scores allows us to distinguish style attributes from content.

2.1.1 Delete Transformer

To build intuition, any attention-based style classifier defines a probability distribution over style labels:

$$p(s|x) = g(v, \alpha) \tag{1}$$

where v is a tensor such that $v[i]$ is an encoding of $x[i]$, and α is a tensor of attention weights such that $\alpha[i]$ is the weight attributed to $v[i]$ by the classifier in deciding probabilities for each s_j. The α scores can be treated as *importance* scores and be used to identify attribute words, (which typically tend to have higher scores). Motivated by the recent successes of the Transformer (Vaswani et al., 2017) and more specifically, BERT (Devlin et al., 2018), on a number of text classification tasks (including achieving state-of-art results on sentiment classification), we use a BERT-based transformer as our style classifier and refer to it as Delete Transformer **(DT)**. However, since **DT** has multiple attention heads and multiple blocks (layers), extracting a single set of attention weights α is a non-trivial task. This is further complicated by the fact that every layer and head encodes different aspects of semantic and linguistic structure (Vig, 2019). We then use a novel method to extract a specific attention head and layer combination that encodes style information and that can be directly used as *importance* scores.

Attribute extraction: We use the same input representation as Figure 3(b) in Devlin et al. (2018) wherein a '[CLS]' token is added before the sentence tokens. Since the softmax classifica-

tion layer is used over the attention stack of the '[CLS]' token in BERT classifiers, the attention weights of other input tokens that correspond to '[CLS]' are of special interest in identifying significant sentence tokens. First, we iterate over each pair $< h, l >$ (head-layer pair) and extract the attention scores for every token w of x as follows:

$$\alpha_{h,l}(w) = softmax_{w \in x}(Q_{h,l,[CLS]} K^T_{h,l,w}) \quad (2)$$

where 'Q' and 'K' carry the same original connotations of query and key vectors as used by Vaswani et al. (2017), in the Transformer as:

$$Att(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V \quad (3)$$

We then remove the top $\gamma|x|$ tokens from x, based on *importance* scores calculated as in Eq. 2. Keeping in line with Feng et al. (2018), we call this removal a 'reduction', and denote the resulting reduced sentence as $x'_{h,l}$. γ is a parameter we tune to each dataset which allows us to control the proportion of words in a sentence to be deleted, and $|x|$ denotes the number of tokens in x. We calculate a score $z(x'_{h,l})$:

$$z(x'_{h,l}) = \frac{p(s|x'_{h,l}) + \lambda}{\sum_{s'} p(s'|x'_{h,l}) + \lambda} \quad (4)$$

where λ is a smoothing parameter, s is the style label assigned maximum probability by the softmax distribution over all styles in the label set S, and $s' = S - \{s\}$. The final pair $< h_s, l_s >$ out of combinations of all heads H and layers L, is obtained by averaging the score in Eq. 4 over a validation set of 'reduced' sentences D'_{val} as follows:

$$(h_s, l_s) = \underset{h \in H, l \in L}{\text{argmin}} \frac{\sum_{x'_{h,l} \in D'_{val}} z(x'_{h,l})}{|D'_{val}|} \quad (5)$$

A 'reduction' of any input sentence x based on $< h_s, l_s >$ gives us x'_{h_s,l_s} which we refer to as the content c. The removed tokens are the attributes a.

Evaluation of Extracted Attributes: We evaluate our Delete method using human evaluation on Amazon Mechanical Turk[3], on which annotators were asked to choose if all the style-related attributes are extracted correctly by our Delete mechanism, and if any non-style attributes are wrongly deleted. We used 200 random sentences from our test set for sentiment transfer, for this evaluation. Our method deleted all style attributes on 89% of examples, and wrongly deleted non-style attributes only 12% of the time. In comparison, the Delete mechanism proposed by Li et al. (2018) deleted all style attributes only 67% of the time, and wrongly deleted non-style attributes over 29% of the time.

[3]https://www.mturk.com/

2.2 Retrieve

We retrieve a sentence from the target style corpus of sentences according to:

$$x^{tgt} = argmin_{x' \in D_{stgt}} d(c_x, c_{x'}) \quad (6)$$

where d is a distance metric, such that contents which are closer according to d will have compatible attributes as they occur in similar contexts. We experiment with multiple retrieval mechanisms, using cosine similarity over different sentence representations: a) TF-IDF weighted, b) Averaged-GloVe over all tokens of a sentence and c) Universal Sentence Encoder (Cer et al., 2018). We obtain best retrieval results using TF-IDF vector similarity.

2.3 Generate

Our approach to generate sentences of the target style leverages both the power of transfer learning by using an unsupervised language model trained on a large corpus of unlabeled text, as well as the Transformer model. The model we use is a multi-layer 'decoder-only' Transformer which is based on the Generative Pre-trained Transformer (GPT) of Radford et al.. This is our Generative Style Transformer **(GST)**. **GST** has masked attention heads that enable it to look only at the tokens to its left, and not to those to its right. **GST** derives inspiration from the fact that recently, many large generatively pre-trained Transformer models have shown state-of-art performance upon being finetuned on a number of downstream tasks. It is trained to learn a representation of content words and (retrieved) attribute words presented to it, and generate fluent sentences in the domain of the target style while attending to both content and attribute words appropriately.

2.3.1 Variants of GST (B-GST and G-GST)

Taking cues from Li et al. (2018), we train GST in two flavors: the Blind Generative Style

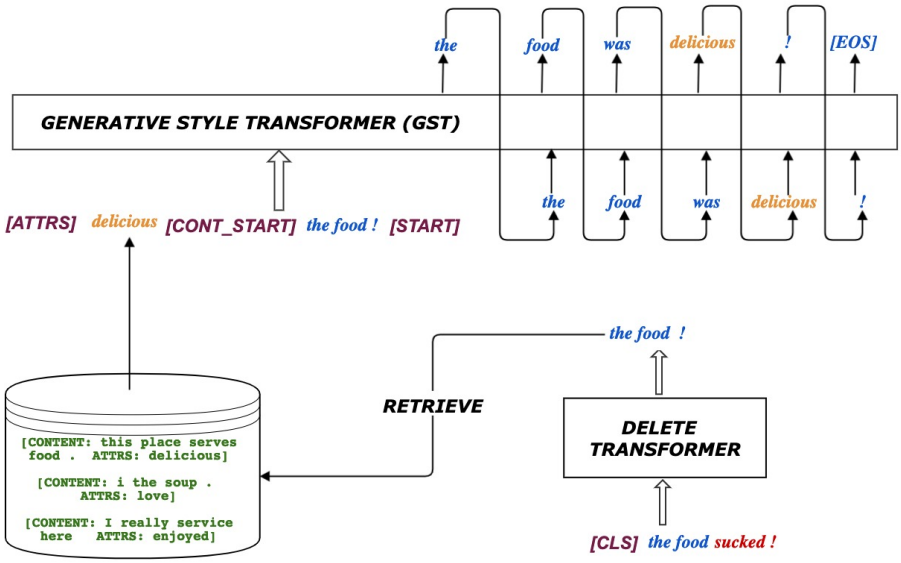

Figure 1: Our architecture, with an example from the Yelp dataset for the task of sentiment transfer

Transformer **(B-GST)** and the Guided Generative Style Transformer **(G-GST)**. For a sentence x of a source style s^{src} with content c and retrieved (target style) attributes a^{tgt}, the two variations are learnt as follows.

B-GST: The inputs to this model are only c, and s^{tgt}. The output y of the model is the generated sentence in the target style. In this setting, the model is free to generate the output sentence, given the content and the target style, and is **blind** to specific desired target attributes. **B-GST** can be useful in cases when the target corpus does not have sentences that are similar to the source corpus, which causes the Retrieve model to retrieve incompatible target attributes.

G-GST: The inputs to this model are c, and a^{tgt}, and the output y of the model is the generated sentence in the target style. In this setting, the model is **guided** towards generating a target sentence with desired attributes. **G-GST** is useful for two reasons. Firstly, in cases when the target corpus has similar sentences to the source corpus, it reduces sparsity by giving the model information of target attributes. Secondly, and more importantly, it allows fine-grained control of output generation by manually specifying target attributes that we desire during inference time, without even using the Retrieve component. This controllability is an important feature of G-GST that most other latent-representation based style transfer approaches do not offer.

2.3.2 Input Representation and Output Decoding

Taking inspiration from Devlin et al. (2018), we add special tokens to indicate target style, and to indicate the demarcation between content and attributes. For **B-GST** the input at timestep t of target sentence prediction consists of special tokens to denote: a) target style s^{tgt}, b) the start of content c, d) the start of output, followed by all target tokens up till and including timestep $t-1$. **G-GST** has a similar input representation, except that a special token to indicate start of retrieved attributes is added, and the retrieved attributes are provided before the content. The target style s^{tgt} is not provided. Our end-to-end architecture for **G-GST** is depicted in Figure 1, including input representation. **B-GST** is similar in nature, except that it does not use a retrieve component. At timestep t, both **GST**s predict the t^{th} output token, by generating a probability distribution over words in the vocabulary according to: a) $p(y_t|c, y_1, y_2, ..y_{t-1})$ for **B-GST**, and b) $p(y_t|c, a^{tgt}, y_1, y_2, ..y_{t-1})$ for **G-GST**. This is done by using a softmax layer over the topmost Transformer block corresponding to y_{t-1}. During training time, we use the 'teacher forcing' or 'guided approach' (Bengio et al., 2015; Williams and Zipser, 1989) over decoded tokens. During test time, we beam search using softmax probabilities with a look-left window of 1 and a beam width of 5. The output beam (out of the top 5 final beams) that obtains the highest target-style match

score using the Delete Transformer described earlier, is chosen as the output sentence.

2.3.3 Training

Since we do not have a parallel corpus, both GSTs are trained to minimize the reconstruction loss. Specifically, for a sentence x, the model learns to reconstruct $y = x$ given c_x, its own attributes a_x (only for **G-GST**) and its own style s^{src} (only for **B-GST**). More formally **B-GST** learns to maximize the following objective:

$$L(\theta) = \sum_{(x,s^{src})\in D} \log p(x|c_x, s^{src}; \theta) \quad (7)$$

However, training **G-GST** using the reconstruction loss in this manner results in the model learning to trivially combine c_x and a_x to generate x back. In reality we want it to be capable of *adapting* target attributes into the context of the source content, in a non-trivial manner to produce a fluent sentence in the target style. To this end, we noise the inputs of the **G-GST** model during training time, by choosing random attributes for 10% of the examples (5% from the source style and 5% from the target style), to replace a_x. Denoting the chosen attribute for an example (either noisy or its own) to be a'_x, **G-GST** learns to maximize the following objective:

$$L(\theta) = \sum_{(x,s^{src})\in D} \log p(x|c_x, a'_x; \theta) \quad (8)$$

2.3.4 Model Details and Pre-training

We use the PyTorch implementation of the pre-trained Transformer by HuggingFace[4], which uses the pre-trained OpenAI GPT model[5]. This model is pre-trained by Radford et al. on the BookCorpus dataset[6] of over 7000 books (around 800M words). **GST** has a sequence length of 512, 12 blocks (or layers), and 12 attention-heads in each block. All internal states (keys, queries, values, word embeddings, positional embeddings) are 768-dimensional. Input text is tokenized using Byte-Pair Encoding (BPE).

3 Experiments

3.1 Datasets

We use 5 different datasets for our experiments. These datasets have been used in previous works on style transfer.

Dataset	Style	Train	Dev	Test
YELP	Positive	270K	2000	500
	Negative	180K	2000	500
CAPTIONS	Romantic	6000	300	0
	Humorous	6000	300	0
	Factual	0	0	300
AMAZON	Positive	277K	985	500
	Negative	279K	1015	500
POLITICAL	Democrat	270K	2000	28K
	Republican	270K	2000	28K
GENDER	Male	1.34M	2250	267K
	Female	1.34M	2250	267K

Table 1: Dataset statistics

We use the YELP, AMAZON and CAPTIONS datasets as used by Li et al. (2018), and we retain the same train-dev-test split that they use. Further, they also provide human gold standard references for the test sets of all 3 of the above. We use the POLITICAL (Voigt et al., 2018) and GENDER (Reddy and Knight, 2016) datasets as used by Prabhumoye et al. (2018). We have retained the same train-dev-test split that they use. Table 1 shows statistics of these datasets. Following are brief descriptions of the datasets:

YELP: Reviews of businesses on Yelp, with each review labelled as having either positive or negative sentiment.

AMAZON: Product reviews on Amazon, with each review labelled as having either positive or negative sentiment.

CAPTIONS: Image captions, with each caption labeled as either factual, romantic, or humorous. The task is to convert factual sentences into romantic and humorous ones.

POLITICAL: Top-level comments on Facebook posts from members of the United States Senate and House who have public Facebook pages, with each comment labelled as having been posted by either a Republican or a Democrat politician.

GENDER: Reviews of food businesses on Yelp, with each review labelled as either of the two genders (male or female) corresponding to markers of sex.

3.2 Comparison to Previous Works

On the Yelp, Amazon, and Captions dataset, we compare with 3 previous adversarially trained models: StyleEmbedding **(SE)** (Fu et al., 2018), MultiDecoder **(MD)** (Fu et al., 2018),

[4]https://github.com/huggingface/pytorch-pretrained-BERT

[5]https://github.com/openai/finetune-transformer-lm

[6]https://www.smashwords.com/

	YELP				AMAZON				CAPTIONS			
Model	Cont.	Flu.	Sty.	All	Cont.	Flu.	Sty.	All	Cont.	Flu.	Sty.	All
D&R	18	14.5	22	13	40	35.5	45	39.5	20.5	24	**47.75**	30.25
B-GST	**66**	**64**	**60**	**69**	**48**	**50.5**	**45.5**	**49**	**65.5**	**56.75**	34	**52.75**
None	16	21.5	18	18	12	14	9.5	11.5	14	19.25	18.5	17

Table 2: Human evaluation results - each cell indicates the percentage of sentences preferred down a column (Cont. = Content preservation ; Flu. = Fluency ; Sty. = Target Style Match ; All = Overall)

	POLITICAL		GENDER	
Model	Cont.	Flu.	Cont.	Flu.
BT	22	29	18	23
B-GST	**69**	**61**	**70**	**67**
None	9	10	12	10

Table 3: Human evaluation results - each cell indicates the percentage of sentences preferred down a column (Cont. = Content preservation ; Flu. = Fluency)

CrossAligned **(CA)** (Shen et al., 2017)) and the 2 best models - DeleteOnly **(D)** and DeleteAndRetrieve **(D&R)** of Li et al. (2018) trained using the DRG framework. A brief description of the first 3, and a detailed description of the last 2 models can be found in Li et al. (2018), so we omit elaborating on them here. At the time of writing this paper, these models are the top performing models on Yelp, Amazon and Captions, with the **D&R** model of Li et al. (2018) showing state-of-art performance. Output sentences of each of these 5 models on fixed test sets, also annotated with human reference gold standards **(H)** on all 3 datasets are provided by Li et al. (2018). We use the same for our comparison and evaluation. On the Political and Gender datasets, we compare our models against that of Prabhumoye et al. (2018), which is the state-of-art on these 2 datasets at the time of writing this paper. Their trained models for both these datasets are made publicly available. They use back-translation **(BT)** using an LSTM as a mechanism to learn latent representations of source sentences, and then employ adversarial generation techniques to make the output match a desired style (Prabhumoye et al., 2018).

4 Evaluation of Results

The widely agreed upon goals for a style transfer system are 1) Content preservation of the non-stylistic parts of the source sentence, 2) Style transfer strength of the stylistic attributes to the target style and 3) Fluency and correct grammar of the generated target sentence (Mir et al., 2019). To this end, we use both human and automatic evaluation to measure model performance.

4.1 Human Evaluation

YELP, AMAZON and CAPTIONS: Li et al. (2018) report state-of-art results which we corroborate through manual and automatic metrics. We then proceed to obtain human evaluations on these models along with ours through Amazon Mechanical Turk[7]. Specifically, we ask annotators to rate each pair of generated sentences given the source sentence, on content preservation, style transfer strength, fluency, and overall success. For each parameter, they are asked to choose which of the generated sentences is better, or neither of the two if they are unable to decide. Table 2 presents results on our best scoring model **B-GST** with the previous best scoring model **D&R** as a percentage of times one was preferred over the other.

POLITICAL and GENDER: On these 2 datasets, (Prabhumoye et al., 2018) report state-of-art results using their model **BT**, which we similarly corroborate. A comparison of our best model **B-GST**, with their results using **BT** is presented in Table 3 as a percentage of times one was preferred over the other. Since judging target style strength on these two tasks are hard for MTurkers, they only rate these datasets for content and fluency.

4.2 Automatic Evaluation

As has been done by previous works, we attempt to use automatic methods of evaluation to assess the performance of different models. To estimate target style strength, we use style classifiers that we train on the same training-dev-test split of Table 1, using FastText[8] (Joulin et al., 2017). These classifiers achieve 98%, 86%, 80%, 92% and 82% accuracies on the test sets of Yelp, Amazon, Captions, Political and Gender respectively. To measure content preservation, we calculate the BLEU score (Papineni et al., 2001) between the gener-

[7]https://www.mturk.com/
[8]https://fasttext.cc/

	YELP				AMAZON				CAPTIONS			
Model	GL	BL_S	PL	AC	GL	BL_S	PL	AC	GL	BL_S	PL	AC
SRC	7.6	100.0	24.0	2.6	19.3	100.0	32.9	20.4	11.3	100.0	34.4	50.0
CA	4.4	48.0	72.8	72.7	0.0	15.2	**30.1**	**83.1**	1.6	24.1	**10.1**	50.8
SE	5.9	78.0	115.9	8.6	0.0	16.7	129.8	45.5	5.9	53.8	80.3	51.0
MD	5.0	57.3	205.6	46.8	0.0	16.5	122.5	71.8	4.5	48.7	40.5	51.3
D	6.4	56.7	75.8	85.0	0.0	16.2	55.0	50.6	7.8	59.1	52.5	57.5
D&R	6.9	58.0	90.0	**89.3**	0.0	16.1	42.2	50.9	7.8	49.1	28.8	**67.5**
G-GST	3.8	70.6	64.4	78.3	13.4	71.0	171.0	57.6	1.1	13.1	45.0	52.3
B-GST	**11.6**	**71.0**	**38.6**	87.3	**14.9**	**73.6**	55.2	60.0	**12.6**	**68.3**	28.9	56.0
H	100.0	58.1	67.2	75.2	100.0	70.5	77.0	42.6	100.0	36.4	41.4	55.5

Table 4: Automatic evaluation results (GL = GLEU, BL_s = BLEU ; PL = Perplexity ; AC = Target Style Accuracy ; SRC = Input Sentence ; B-GST and G-GST are our models ; H = Human Reference)

	POLITICAL			GENDER		
Model	BL_S	PL	AC	BL_S	PL	AC
SRC	100.0	62.9	9	100	183.4	18.9
BT	40.2	**61.9**	**88.0**	46.0	196.2	52.9
G-GST	76.7	241.6	67.4	78.5	252.0	49.0
B-GST	**79.2**	104.4	71.2	**82.5**	**189.2**	**57.9**

Table 5: Automatic evaluation results (BL_s = BLEU; PL = Perplexity; AC = Target Style Accuracy; SRC = Input Sentence; B-GST and G-GST are our models)

ated and source sentences. To measure fluency, we finetune a large pre-trained language model, OpenAI GPT-2 (note that this is different from GPT-1 on which our Generate model is based) on the target sentences using the same training-dev-test split of Table 1. We use this language model to measure perplexity of generated sentences. The language models achieve perplexities of 24, 33, 34, 63 and 81 on the test sets of Yelp, Amazon, Captions, Political and Gender respectively. As we analyze in the next section, automatic metrics are inadequate at measuring the success of a good style transfer system.

GLEU: As a step towards finding an automatic metric that compares with human judgements, we propose the use of the Generalized Language Evaluation Understanding Metric (GLEU) (Napoles et al., 2015) - originally proposed as a grammatical error correction (GEC) metric. In the interest of space, we omit writing the elaborate equations and explanation for GLEU in this paper, but instead point the reader to Section 4 of Napoles et al. (2015) for the same. The formulation of GEC is quite similar to our formulation of style transfer in that style transfer involves making localized edits to the input sentence. Unlike BLEU, which takes only the target reference and the generated output into consideration, GLEU considers both of these as well as the source sentence too. It is a suitable metric for style transfer because it a) penalizes words of the source that were wrongly changed in the generated sentence, b) rewards words that were successfully changed and c) rewards those that were successfully retained from the source sentence to match those in the reference sentence. We use the implementation of GLEU[9] provided by Napoles et al. (2015).

Tables 4 and 5 show a comparison of automatic metrics between our models and previous models described earlier.

4.3 Result Analysis

From human evaluations in Tables 2 and 3, we see that our models (specifically, **B-GST**) outperform state-of-art systems by a good margin on almost all parameters as judged by humans, across all datasets. More importantly, as Table 6 shows, our models generate realistic and natural-sounding sentences while retaining core content - an aspect on which previous models seem to be seriously lacking. While our **G-GST** model does worse than **B-GST** due to a weak Retrieve mechanism, **G-GST** provides us a way to guide the generation and control attributes, making it more suitable for real-world applications after improving Retrieve in future. We find that metrics based on learned models - perplexity and accuracy, do not correlate entirely well with human evaluations, an observation also shared by Li et al. (2018). They are also heavily dependant on the distribution of data that they are trained on. A system that simply chooses a random sentence from the target training corpus as its output will score highly on both these metrics. For instance, the **BT** model in Table 5 has a high style but a considerably lower BLEU

[9]https://github.com/cnap/gec-ranking

Example #1	**YELP (Positive to Negative)**	**YELP (Negative to Positive)**
SRC	i love this place , the service is always great !	the store is dumpy looking and management needs to change .
SE	i love this place , the service is always great !	the store is bought the building does n't deal .
D	i paid _num_ minutes before the gifted , not , a huge plus , n't .	the store is dumpy looking and management is fantastic and needs to change .
D&R	i did not like the homework of lasagna , not like it , .	the store is clean and well dumpy looking and management needs to change .
G-GST	**i used this place , the service is always awful !**	**the store is looking and management is excellent to .**
B-GST	**i hate this place , the service is always terrible !**	**the store is looking great and management to perfection .**
Example #2	**AMAZON (Positive to Negative)**	**AMAZON (Negative to Positive)**
SRC	i finally made he purchase and am glad i did .	i m just looking forward to the day i get to replace it .
SE	i finally made he purchase and am glad i did .	i m just looking forward to the right away i get it for it .
D	i finally made it and was excited to purchase and am glad i did .	i m just looking forward to the day i get to replace my old one .
D&R	i finally made i will try another purchase and am glad i did .	looking forward to using it on turkey day ! .
G-GST	**i finally made he purchase and am embarrassed i smell pungent .**	**i m looking to the same day i get to use it .**
B-GST	**i finally made him purchase and am sorry i did .**	**i m looking forward to using the day i get to use it .**
Example #3	**CAPTIONS (Factual to Romantic)**	**CAPTIONS (Factual to Humorous)**
SRC	people gather around a life size chess game .	three brown and black dogs are splashing in the water .
SE	people gather at a red wooden advertisement of players enjoy .	three small and brown dog are splashing in the water .
D	two young boys have working around a dream line and dream of childhood.	three black and brown dogs are sitting in the water to search of fish .
D&R	people gather around a carnival event , all determined to win the game .	two black and brown dogs are running in the water like a fish.
G-GST	**people gather around a life size chess game to celebrate life ' s happiness .**	**three brown and black dogs are splashing in the water talking to each other .**
B-GST	**two people gather around a life size chess game to celebrate life .**	**three brown and black dogs are splashing in the water looking for mermaids .**
Example #4	**POLITICAL (Democrat to Republican)**	**POLITICAL (Republican to Democrat)**
SRC	thank you for your commitment to a strong public education system , senator !	i absolutely agree with senator paul's actions .
BT	thanks for your vote for a balanced budget amendment , sir !	i ' m merchandising with the rhetoric of senator warren .
G-GST	**thank you for your commitment to a strong conservative system , governor !**	**i absolutely agree with brian ' s dire actions . .**
B-GST	**thank you for your commitment to a strong constitutional system , senator scott !**	**i absolutely agree with elizabeth warren ' s actions .**
Example #5	**GENDER (Male to Female)**	**GENDER (Female to Male)**
SRC	this is a spot that ' s making very solid food , with good quality product .	this a great place for a special date or to take someone from out of town .
B-GST	**this is a cute spot that ' s making me very happy , with good quality product .**	**this a great place for a bachelor or to meet someone from out of town .**

Table 6: Examples of generated sentences to be compared down a column (B-GST and G-GST are our models, SRC is the input sentence). Attributes are colored.

score than **B-GST**. It is important therefore, to not consider them in isolation. Further, human reference sentences themselves score poorly using both these metrics as shown in these tables. Manual inspection of classifier accuracies shows that these classifiers give unreliable outputs that do not match human ratings. This is the case with the **CA** model in Table 2. Similar problems exist with regarding BLEU in isolation. A system that simply copies the source sentence will obtain high BLEU scores.

GLEU, however seems to strike a balance between target style match and content retention, as it takes the source, reference as well as predicted sentence into account. We see that GLEU scores also correlate with our own human evaluations as well as those of Li et al. (2018). While a detailed statistical correlation study is left for future work, the fact remains that GLEU is not susceptible to the weaknesses of other automatic metrics described above. Our uniformly state-of-art GLEU scores possibly indicate that we make only necessary edits to the source sentence.

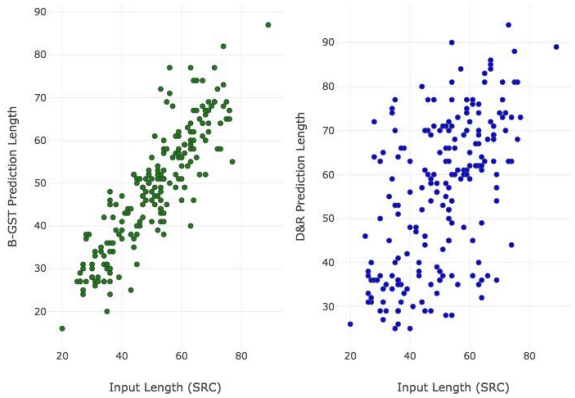

Figure 2: Correlation of B-GST (ours, left) with input sentence lengths vs D&R's (right) sentence lengths with input sentence lengths.

Keeping all the above considerations in mind, automatic metrics are still indicative and useful as they can be scaled to evaluate larger sets of models and datasets. From Tables 4 and 5, we see that we consistently outperform current state-of-art systems on BLEU. As shown by our high BLEU scores, one can conclude to some extent that our models retain non-stylistic parts well. Figure 2 shows that unlike the current state-of-art **D&R** model, the lengths of our generated sentences closely correlate with source sentence lengths. **B-GST** scores well on perplexity across datasets, a consistency that is not exhibited by any other model.

5 Related Work

One category of previous approaches is based on training adversarial networks to learn a latent representation of content and style. Shen et al. (2017) train a cross-aligned auto-encoder, with a shared content and separate style distribution. Hu et al. (2017) use VAEs with attribute discriminators to learn similar latent representations. This approach has been later encapsulated in encoder-decoder frameworks (Fu et al., 2018; John et al., 2018; Zhang et al., 2018a,b). Problems with these approaches have been discussed in the introduction.

Approaches that do not rely on a latent representation to separate content and attribute exist too. These include reinforcement learning based approaches (Xu et al., 2018; Gong et al., 2019), an unsupervised machine translation based approach (Subramanian et al., 2018) and the DRG approach (Li et al., 2018). The former two approaches suffer from sparsity and convergence issues and hence generate sentences of low-quality.

Previous approaches to use attention weights to extract attribute significance exist (Feng et al., 2018; Li et al., 2016; Globerson and Roweis, 2006), including the salience deletion method of Li et al. (2018) but they do not perform well on understanding sentence context while choosing attributes, and do not leverage the contextual capacity of a Transformer. Lastly, Dai et al. (2019) describe the use of Transformers for style transfer in an adversarial generator-discriminator setting, by adding an additional style embedding to the transformer. We are unable to do a comparitive study as they do not yet publish their code or outputs. The same is the case for Subramanian et al. (2018).

6 Conclusion

We propose the Generative Style Transformer that outperforms state-of-art systems on sentiment, gender and political slant. Our model leverages the DRG framework, massively pre-trained language models and the Transformer network itself.

Acknowledgments

The authors of this paper would like to thank Swati Tiwari, Nishant Thakur and Akshit Mittal for their help with evaluation of results, and the anonymous reviewers for their suggestions and comments.

References

Samy Bengio, Oriol Vinyals, Navdeep Jaitly, and Noam Shazeer. 2015. Scheduled sampling for sequence prediction with recurrent neural networks. In *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'15, pages 1171–1179, Cambridge, MA, USA. MIT Press.

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.

Ning Dai, Jianze Liang, Xipeng Qiu, and Xuanjing Huang. 2019. Style transformer: Unpaired text style transfer without disentangled latent representation.

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*.

Shi Feng, Eric Wallace, Alvin Grissom II, Mohit Iyyer, Pedro Rodriguez, and Jordan Boyd-Graber. 2018. Pathologies of neural models make interpretations difficult. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3719–3728, Brussels, Belgium. Association for Computational Linguistics.

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*.

Amir Globerson and Sam Roweis. 2006. Nightmare at test time: Robust learning by feature deletion. In *Proceedings of the 23rd International Conference on Machine Learning*, ICML '06, pages 353–360, New York, NY, USA. ACM.

Hongyu Gong, Suma Bhat, Lingfei Wu, Jinjun Xiong, and Wen mei Hwu. 2019. Reinforcement learning based text style transfer without parallel training corpus.

Zhiting Hu, Zichao Yang, Xiaodan Liang, Ruslan Salakhutdinov, and Eric P. Xing. 2017. Toward controlled generation of text. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 1587–1596, International Convention Centre, Sydney, Australia. PMLR.

Vineet John, Lili Mou, Hareesh Bahuleyan, and Olga Vechtomova. 2018. Disentangled representation learning for non-parallel text style transfer.

Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. 2017. Bag of tricks for efficient text classification. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 427–431. Association for Computational Linguistics.

Jiwei Li, Will Monroe, and Daniel Jurafsky. 2016. Understanding neural networks through representation erasure. *CoRR*, abs/1612.08220.

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.

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

Courtney Napoles, Keisuke Sakaguchi, Matt Post, and Joel Tetreault. 2015. Ground truth for grammatical error correction metrics. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 588–593, Beijing, China. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2001. Bleu: a method for automatic evaluation of machine translation. In *ACL*.

Shrimai Prabhumoye, Yulia Tsvetkov, Ruslan Salakhutdinov, and Alan W Black. 2018. Style transfer through back-translation. In *Proc. ACL*.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. Improving language understanding by generative pre-training.

Sudha Rao and Joel Tetreault. 2018. Dear sir or madam, may I introduce the GYAFC dataset: Corpus, benchmarks and metrics for formality style transfer. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 129–140, New Orleans, Louisiana. Association for Computational Linguistics.

Sravana Reddy and Kevin Knight. 2016. Obfuscating gender in social media writing. In *Proceedings of the First Workshop on NLP and Computational Social Science*, pages 17–26, Austin, Texas. Association for Computational Linguistics.

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

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

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 5998–6008. Curran Associates, Inc.

Jesse Vig. 2019. Visualizing attention in transformer-based language representation models.

Rob Voigt, David Jurgens, Vinodkumar Prabhakaran, Dan Jurafsky, and Yulia Tsvetkov. 2018. RtGender: A corpus for studying differential responses to gender. In *Proc. LREC*.

Ronald J. Williams and David Zipser. 1989. A learning algorithm for continually running fully recurrent neural networks.

Jingjing Xu, Xu SUN, Qi Zeng, Xiaodong Zhang, Xuancheng Ren, Houfeng Wang, and Wenjie Li. 2018. Unpaired sentiment-to-sentiment translation: A cycled reinforcement learning approach. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 979–988, Melbourne, Australia. Association for Computational Linguistics.

Cheng Yang, Maosong Sun, Xiaoyuan Yi, and Wenhao Li. 2018. Stylistic Chinese poetry generation via unsupervised style disentanglement. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3960–3969, Brussels, Belgium. Association for Computational Linguistics.

Ye Zhang, Nan Ding, and Radu Soricut. 2018a. SHAPED: Shared-private encoder-decoder for text style adaptation. 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 1528–1538, New Orleans, Louisiana. Association for Computational Linguistics.

Zhirui Zhang, Shuo Ren, Shujie Liu, Jianyong Wang, Peng Chen, Mu Li, Ming Zhou, and Enhong Chen. 2018b. Style transfer as unsupervised machine translation. *CoRR*, abs/1808.07894.

Ganbin Zhou, Ping Luo, Rongyu Cao, Fen Lin, Bo Chen, and Qing He. 2017. Mechanism-aware neural machine for dialogue response generation. In *Thirty-First AAAI Conference on Artificial Intelligence*.