Text Style Transfer

What is style of language?

styles	sentences
Sentiment	**Positive**: great food but horrible staff and very very rude workers! **Negetive**: great food , awesome staff , very personableand very efficient atmosphere !
Humorous	**Factual**: a black and white dog is running through shallow water Humorous:a black and white dog is running through shallow water like a fish
Formality	Informal: Wow , I am very dumb in my observation skills ..... Formal:I do not have good observation skills .
...	

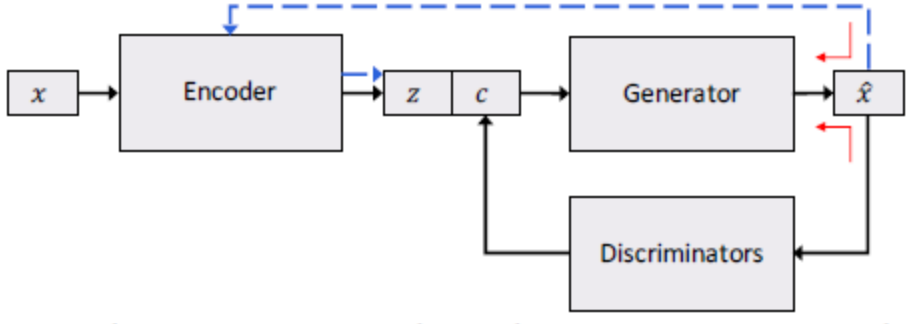
Hu et al. Toward Controlled Genetation of Text. ICML2017.

Main Idea:

1.Latent code represents content and attribute

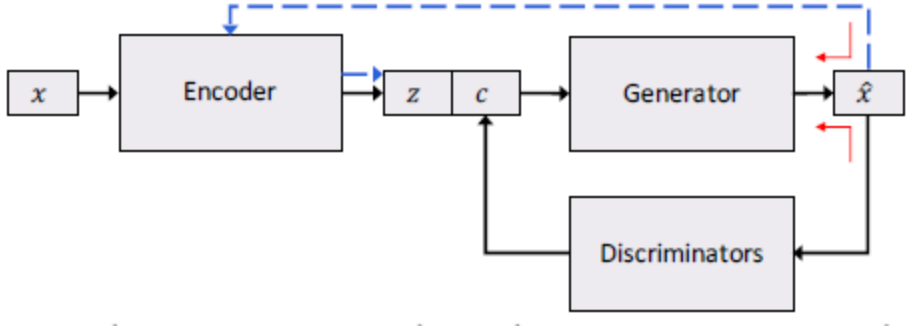
2.VAE with a discriminator of attributes

Model Details



1. $\hat{x} \sim G(z, c) = p_G(\hat{z}|z, c) = \prod_t p(\hat{x}_t|\hat{x}^{<t}, z, c)$
2. $z \sim E(x) = q_E(z|x)$
3. $\mathcal{L}_{VAE}(\theta_G, \theta_E; x) = KL(q_E(z|x)||p(z)) - \mathbb{E}_{q_E(z|x)q_D(c|x)}[\log p_G(x|z, c)]$
4. $D(x) = q_D(c|x)$

Model Details



5. $\mathcal{L}_{Attr,c}(\theta_G) = -\mathbb{E}_{p(x)p(c)}[\log q_D(c|\tilde{G}_\tau(z,c))]$
6. $\mathcal{L}_{Attr,c}(\theta_G) = -\mathbb{E}_{p(x)p(c)}[\log q_E(c|\tilde{G}_\tau(z,c))]$
7. $min\theta_G\mathcal{L}_G = \mathcal{L}_{VAE} + \lambda_c\mathcal{L}_{Attr,c} + \lambda_z\mathcal{L}_{Attr,z}$

Datasets

1.**Sentiment**：1)Stanford Sentiment Treebank. 2)Lexicon. 3)IMDB

2.**Tense**

Result

Model	Dataset		
	SST-full	SST-small	Lexicon
S-VAE	0.822	0.679	0.660
Ours	**0.851**	**0.707**	**0.701**

Table 1. Sentiment accuracy of generated sentences. S-VAE (Kingma et al., 2014) and our model are trained on the three sentiment datasets and generate 30K sentences, respectively.

Varying the code of tense	
i thought the movie was too bland and too much	this was one of the outstanding thrillers of the last decade
i guess the movie is too bland and too much	this is one of the outstanding thrillers of the all time
i guess the film will have been too bland	this will be one of the great thrillers of the all time

Table 3. Each triple of sentences is generated by varying the tense code while fixing the sentiment code and z.

Failure cases	
the plot is not so original	it does n't get any better the other dance movies
the plot weaves us into <unk>	it does n't reach them , but the stories look
he is a horrible actor 's most part	i just think so
he 's a better actor than a standup	i just think !

Table 5. Failure cases when varying sentiment code with other codes fixed.

Shen et al. Style transfer from non-parallel text by cross-alignment. NIPS2017.

Main idea:

1. non-parallel corpora
2. preserve the content of the source sentence but render the sentence consistent with desired presentation constraints

Formulation

1. a latent style variable y is generated from some distribution $p(y)$;
2. a latent content variable z is generated from some distribution $p(z)$;
3. a datapoint x is generated from conditional distribution $p(x|y,z)$.
4. $p(x_1|x_2;y_1,y_2)$ or $p(x_2|x_1;y_1,y_2)$
5. $p(x_1|x_2;y_1,y_2)=\int_z p(x_1,z|x_2;y_1,y_2)dz$
   $=\int_z p(z|x_2,y_2)\cdot p(x_1|y_1,z)dz$
   $=\mathbb{E}_{z\sim p(z|z_2,y_2)}[p(x_1|y_1,z)]$

Model

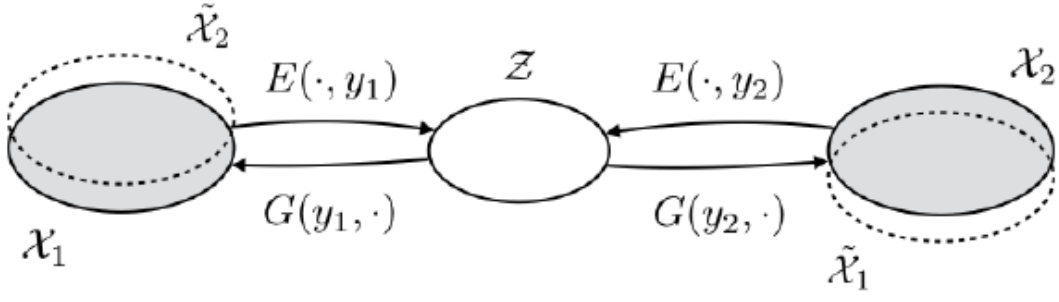


Figure 1: An overview of the proposed cross-alignment method. $\mathcal{X}_1$ and $\mathcal{X}_2$ are two sentence domains with different styles y_1 and y_2, and $\mathcal{Z}$ is the shared latent content space. Encoder E maps a sentence to its content representation, and generator G generates the sentence back when combining with the original style. When combining with a different style, transferred $\tilde{\mathcal{X}}_1$ is aligned with $\mathcal{X}_2$ and $\tilde{\mathcal{X}}_2$ is aligned with $\mathcal{X}_1$ at the distributional level.

$$\begin{aligned}\mathcal{L}_{rec}(\theta_E, \theta_G) =& \mathbb{E}_{x_1 \sim X_1}[-\log p_G(x_1|y_1, E(x_1, y_1))] + \\ & \mathbb{E}_{x_1 \sim X_1}[-\log p_G(x_2|y_2, E(x_2, y_2))] \\ \mathcal{L}_{KL}(\theta) =& \mathbb{E}_{x_1 \sim X_1}[D_{KL}(p_E(z|x_1, y_1)||p(z))] + \\ & \mathbb{E}_{x_2 \sim X_2}[D_{KL}(p_E(z|x_2, y_2)||p(z))]\end{aligned}$$

Aligned auto-encoder

$$\mathcal{L}_{adv}(\theta_E, \theta_D) = \mathbb{E}_{x_1 \sim X_1}[-\log D(E(x_1, y_1))] + \mathbb{E}_{x_2 \sim X_2}[-\log 1 - D(E(x_1, y_1))]$$

$$MIN_{E,G} MAX_D \mathcal{L}_{rec} - \lambda \mathcal{L}_{adv}$$

Cross-aligned auto-encoder

Two discriminators D_1 and D_2, D_1's job is to distinguish between real x_1 and transferred x_2, and D_2's job is to distinguish between real x_2 and transferred x_1.

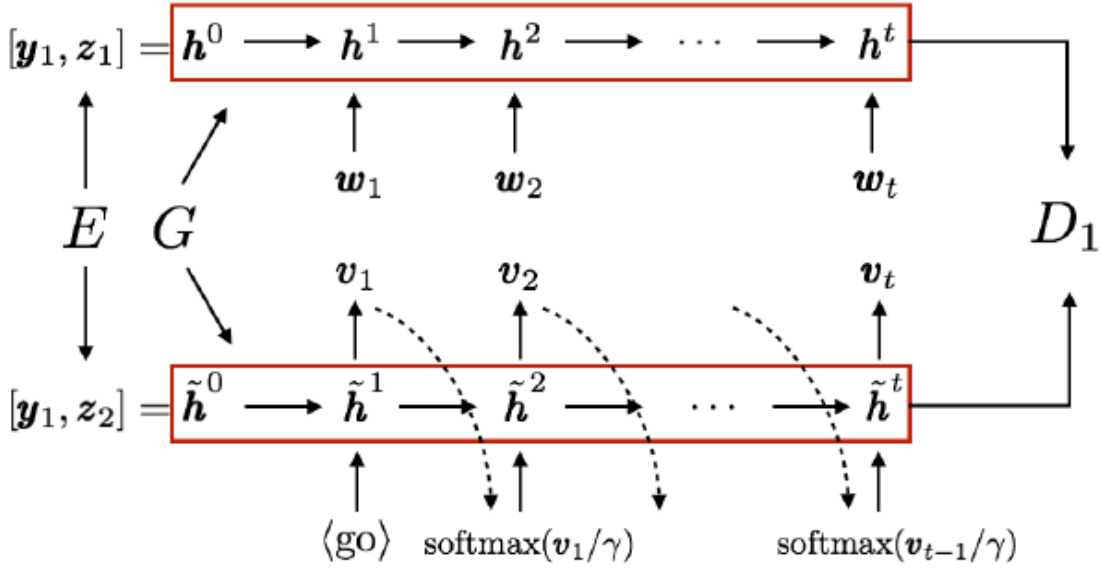


Figure 2: Cross-aligning between x_1 and transferred x_2. For x_1, G is teacher-forced by its words $w_1 w_2 \cdots w_t$. For transfered x_2, G is self-fed by previous output logits. The sequence of hidden states $h^0, \cdots, h^t$ and $\tilde{h}^0, \cdots, \tilde{h}^t$ are passed to discriminator D_1 to be aligned. Note that our first variant aligned auto-encoder is a special case of this, where only h^0 and $\tilde{h}^0$, i.e. z_1 and z_2, are aligned.

Tasks

1.Sentiment modification
2.Word subsititution decipherment
3.Word orfer recovery

Result

Method	accuracy
Hu et al. (2017)	83.5
Variational auto-encoder	23.2
Aligned auto-encoder	48.3
Cross-aligned auto-encoder	78.4

Table 1: Sentiment accuracy of transferred sentences, as measured by a pretrained classifier.

Method	sentiment	fluency	overall transfer
Hu et al. (2017)	70.8	3.2	41.0
Cross-align	62.6	2.8	41.5

Table 2: Human evaluations on sentiment, fluency and overall transfer quality. Fluency rating is from 1 (unreadable) to 4 (perfect). Overall transfer quality is evaluated in a comparative manner, where the judge is shown a source sentence and two transferred sentences, and decides whether they are both good, both bad, or one is better.

Result

From negative to positive
consistently slow . consistently good . consistently fast .
my goodness it was so gross . my husband 's steak was phenomenal . my goodness was so awesome .
it was super dry and had a weird taste to the entire slice . it was a great meal and the tacos were very kind of good . it was super flavorful and had a nice texture of the whole side .
From positive to negative
i love the ladies here ! i avoid all the time ! i hate the doctor here !
my appetizer was also very good and unique . my bf was n't too pleased with the beans . my appetizer was also very cold and not fresh whatsoever .
came here with my wife and her grandmother ! came here with my wife and hated her ! came here with my wife and her son .

Table 3: Sentiment transfer samples. The first line is an input sentence, the second and third lines are the generated sentences after sentiment transfer by Hu et al. (2017) and our cross-aligned auto-encoder, respectively.

Fu et al. Style transfer in text: Exploration and evaluation. AAAI2018.

Main ideas：

1. Challenge: non-parallel、separate、evalution
2. Two general evalution metrics : transfer strength and content preservation
3. Propose two models

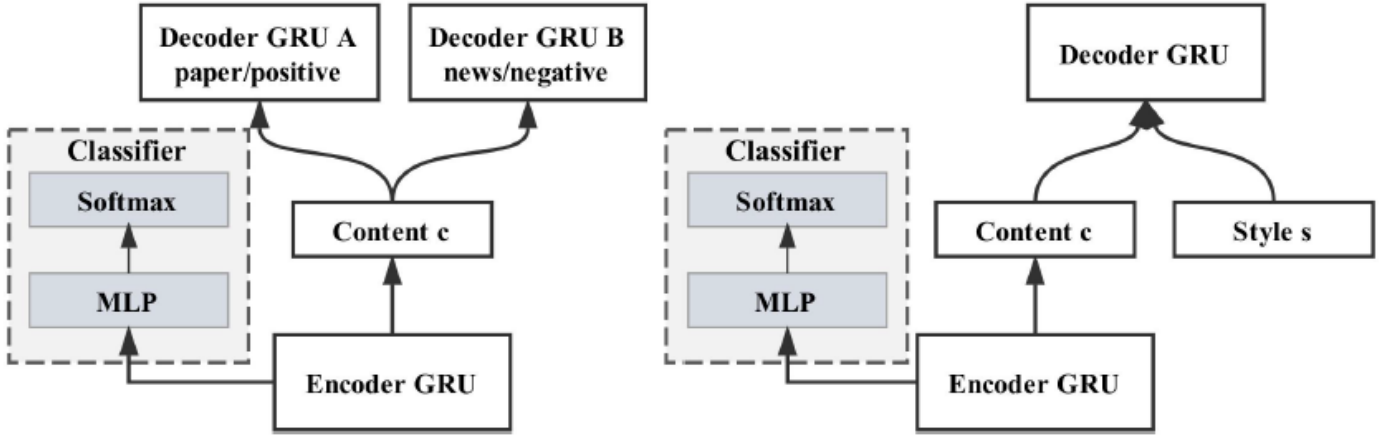


Figure 1: Two models in this paper, multi-decoder (left) and style-embedding (right). Content c represents output of the encoder. Multi-layer Perceptron (MLP) and Softmax constitute the classifier. This classifier aims at distinguishing the style of input X. An adversarial network is used to make sure content c does not have style representation. In style-embedding, content c and style embedding s are concatenated and $[c, e]$ is fed into decoder GRU.

result

source	positive: all came well sharpened and ready to go .
auto-encoder:	→negative: all came well sharpened and ready to go .
multi-decoder:	→negative: all came around , they did not work .
style-embedding:	→negative: my ⟨NUM⟩ and still never cut down it .
source	negative: my husband said it was obvious so i had to return it .
auto-encoder:	→positive: my husband said it was obvious so i had to return it .
multi-decoder:	→positive: my husband was no problems with this because i had to use .
style-embedding:	→positive: my husband said it was not damaged from i would pass right .
source	paper: an efficient and integrated algorithm for video enhancement in challenging lighting conditions
auto-encoder:	→news: an efficient and integrated algorithm for video enhancement in challenging lighting conditions
multi-decoder:	→news: an efficient and integrated and google smartphone for conflict roku together wrong
style-embedding:	→news: an efficient and integrated algorithm, for video enhancement in challenging power worldwide
source	news: luxury fashion takes on fitness technology
auto-encoder:	→paper: luxury fashion takes on fitness technology
multi-decoder:	→paper: foreign banking carbon on fitness technology
style-embedding:	→paper: luxury fashion algorithms on fitness technology

Table 2: Case study of style transfer

Evalutions

Transfer Strength

$$l_{style} = \begin{cases} paper(positive) & output \leq 0.5 \\ news(negative) & output > 0.5 \end{cases}$$

Content preservation

$$\begin{aligned} v_{min}[i] &= \min\{w_1[i], \dots w_n[i]\} \\ v_{mean}[i] &= \text{mean}\{w_1[i], \dots w_n[i]\} \\ v_{max}[i] &= \max\{w_1[i], \dots w_n[i]\} \\ v &= [v_{min}, v_{mean}, v_{max}] \\ score &= \frac{v_s^\top v_t}{\|v_s\| \cdot \|v_t\|} \\ score_{total} &= \sum_{i=1}^{M_{test}} score_i \end{aligned}$$

Xu et al.Unpaired Sentiment-to-Sentiment Translation: A Cycled Reinforcement Learning Approach. ACL2018

Main ideas：

1. Focus on content preservation
2. Using evalution result as rewards in RL

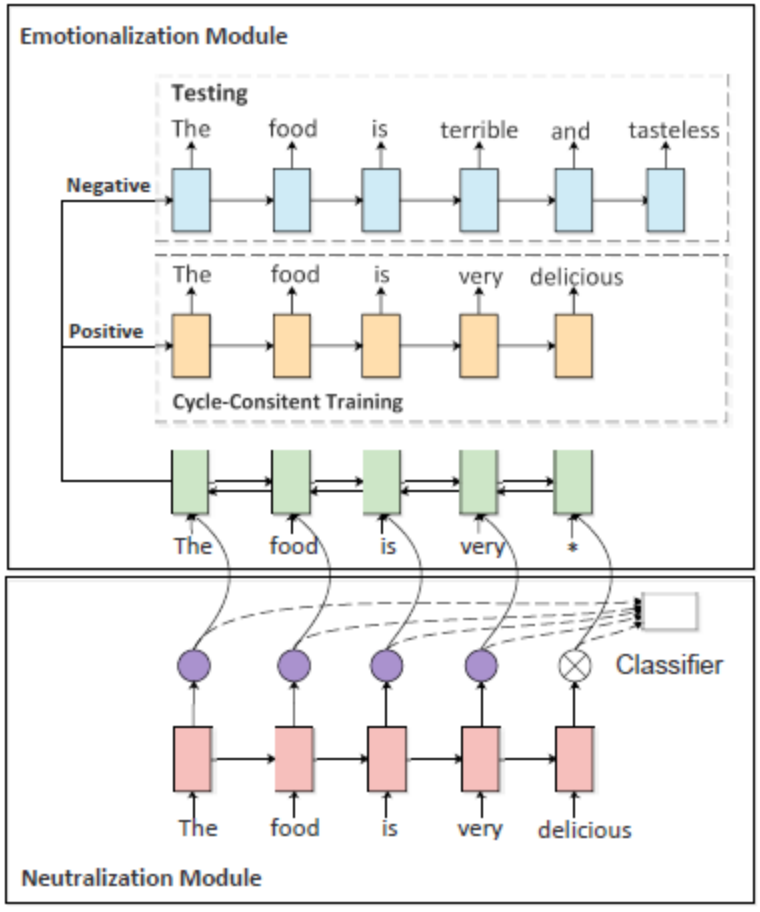


Figure 1: An illustration of the two modules. Lower: The neutralization module removes emotional words and extracts non-emotional semantic information. Upper: The emotionalization module adds sentiment to the semantic content. The proposed self-attention based sentiment classifier is used to guide the pre-training.

Neutralization Module

The neutralization module N_θ is used for explicitly identifying non-emotional words and filtering out emotional information.

1. A single LSTM Network to generate the probability of being neural or being polar for every word in a sentence.
2. Using a sentiment classification to pre-train Neutralization Module

Emotionalization Module

The emotionalization module E_ϕ is responsible for adding sentiment to the neutralized semantic content.

1. Two decoders based on seq2seq model
2. Using pairs of neutraliztion's input/output to pre-train the emotionalization module

Cycled Reinforcement Learning

1. Neutralization and emotionalization can be viewed as the first and second agent respectively.
2. In cycled training, the original sentence can be viewed as the supervision for training the second agent. The gradient for second agent is

$$\nabla_\phi J(\phi) = \nabla_\phi \log(P_{E_\phi}(x|\hat{x}, s))$$

3. According to the policy gradient theorem, the gradient for first agent is

$$\nabla_\theta(\theta) = \mathbb{E}[R_c \cdot \nabla_\theta log(P_{N_\theta}(\hat{\alpha}|x))]$$

Reward

Two parts: Sentiment confident and BLEU.

$$R = (1 + \beta^2) \frac{2 \cdot BLEU \cdot Confid}{(\beta^2 \cdot BLEU) + Confid}$$

Datesets

1. **Yelp Review Dataset**
2. **Amazon Food Review Dataset**

Result

Yelp	ACC	BLEU	G-score
CAAE (Shen et al., 2017)	93.22	1.17	10.44
MDAL (Fu et al., 2018)	85.65	1.64	11.85
Proposed Method	80.00	22.46	**42.38**
Amazon	ACC	BLEU	G-score
CAAE (Shen et al., 2017)	84.19	0.56	6.87
MDAL (Fu et al., 2018)	70.50	0.27	4.36
Proposed Method	70.37	14.06	**31.45**

Table 1: Automatic evaluations of the proposed method and baselines. ACC evaluates sentiment transformation. BLEU evaluates content preservation. G-score is the geometric mean of ACC and BLEU.

Yelp	Sentiment	Semantic	G-score
CAAE (Shen et al., 2017)	7.67	3.87	5.45
MDAL (Fu et al., 2018)	7.12	3.68	5.12
Proposed Method	6.99	5.08	**5.96**
Amazon	Sentiment	Semantic	G-score
CAAE (Shen et al., 2017)	8.61	3.15	5.21
MDAL (Fu et al., 2018)	7.93	3.22	5.05
Proposed Method	7.92	4.67	**6.08**

Table 2: Human evaluations of the proposed method and baselines. *Sentiment* evaluates sentiment transformation. *Semantic* evaluates content preservation.

Li et al. Delete, Retrieve, Generate: A Simple Approach to Sentiment and Style Transfer. NAACL2018.

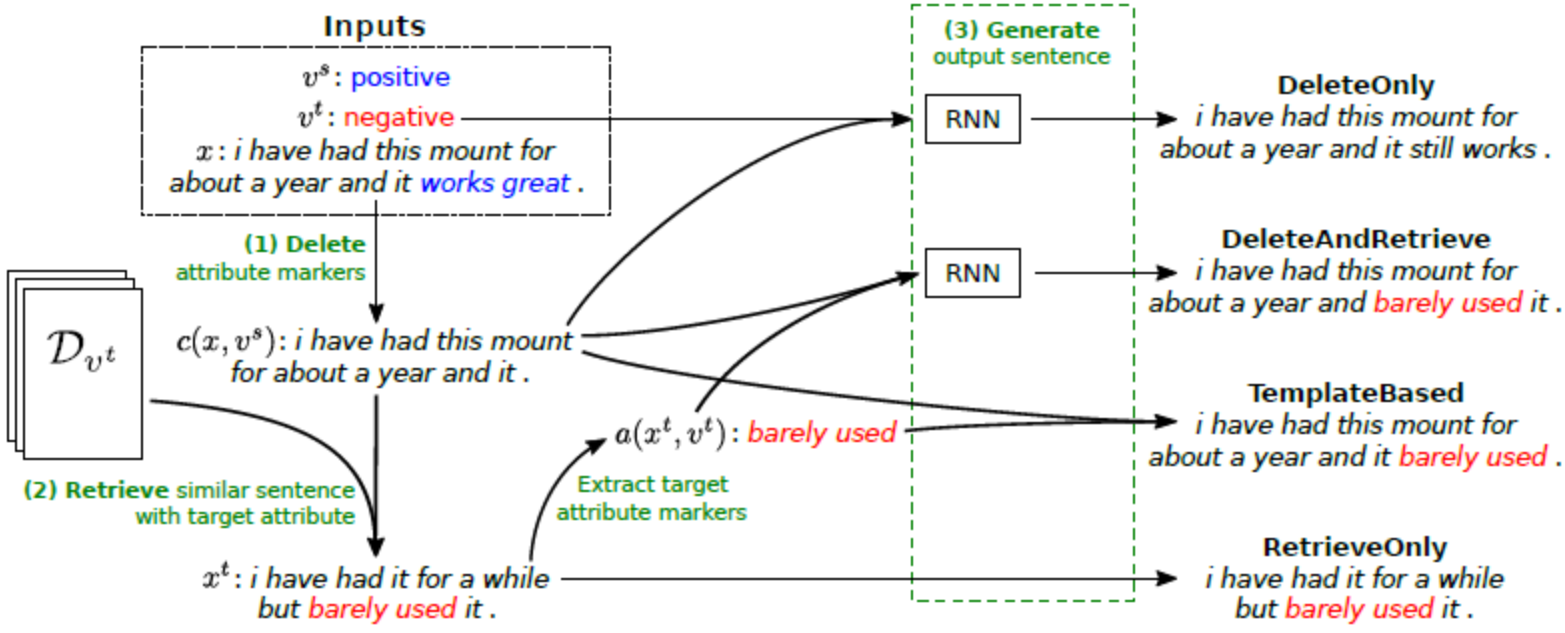


Figure 2: Our four proposed methods on the same sentence, taken from the AMAZON dataset. Every method uses the same procedure (1) to separate attribute and content by deleting attribute markers; they differ in the construction of the target sentence. RETRIEVEONLY directly returns the sentence retrieved in (2). TEMPLATEBASED combines the content with the target attribute markers in the retrieved sentence by slot filling. DELETEANDRETRIEVE generates the output from the content and the retrieved target attribute markers with an RNN. DELETEONLY generates the output from the content and the target attribute with an RNN.

Delete

$$s(u, v) = \frac{\text{count}(u, \mathcal{D}_v) + \lambda}{\left(\sum_{v' \in \mathcal{V}, v' \neq v} \text{count}(u, \mathcal{D}_{v'})\right) + \lambda}$$

For example, for “The chicken was delicious,” we would delete “delicious” and consider “The chicken was. . . ” to be the content

Retrive

$$x^{\text{tgt}} = \underset{x' \in \mathcal{D}_{v^{\text{tgt}}}}{\operatorname{argmin}} d(c(x, v^{\text{src}}), c(x', v^{\text{tgt}})),$$

d:

1. TF-IDF weighted word overlap and
2. Euclidean distance using the content embeddings

Generate

1. RETRIEVEONLY
2. TEMPLATEBASED
3. DELETEONLY
4. DELETEANDRETRIEVE

Result

	YELP				AMAZON				CAPTIONS			
	Gra	Con	Att	Suc	Gra	Con	Att	Suc	Gra	Con	Att	Suc
CROSSALIGNED	2.8	2.9	3.5	14%	3.2	2.5	2.9	7%	3.9	2.0	3.2	16%
STYLEEMBEDDING	3.5	3.7	2.1	9%	3.2	2.9	2.8	11%	3.3	2.9	3.0	17%
MULTIDECODER	2.8	3.1	3.0	8%	3.0	2.6	2.8	7%	3.4	2.8	3.2	18%
RETRIEVEONLY	**4.2**	2.7	**4.2**	25%	3.8	2.8	3.1	17%	**4.2**	2.6	3.8	27%
TEMPLATEBASED	3.0	**3.9**	3.9	21%	3.4	3.6	3.1	19%	3.3	**4.1**	3.5	33%
DELETEONLY	3.0	3.7	3.9	24%	3.7	**3.8**	3.2	24%	3.6	3.5	3.5	32%
DELETEANDRETRIEVE	3.3	3.7	4.0	**29%**	**3.9**	3.7	**3.4**	**29%**	3.8	3.5	**3.9**	**43%**
Human	4.6	4.5	4.5	75%	4.2	4.0	3.7	44%	4.3	3.9	4.0	56%

Table 2: Human evaluation results on all three datasets. We show average human ratings for grammaticality (Gra), content preservation (Con), and target attribute match (Att) on a 1 to 5 Likert scale, as well as overall success rate (Suc). On all three datasets, DELETEANDRETRIEVE is the best overall system, and all four of our methods outperform previous work.

	YELP		CAPTIONS		AMAZON	
	Classifier	BLEU	Classifier	BLEU	Classifier	BLEU
CROSSALIGNED	73.7%	3.1	74.3%	0.1	**74.1%**	0.4
STYLEEMBEDDING	8.7%	**11.8**	54.7%	6.7	43.3%	10.0
MULTIDECODER	47.6%	7.1	68.5%	4.6	68.3%	5.0
TEMPLATEBASED	81.7%	**11.8**	92.5%	**17.1**	68.7%	**27.1**
RETRIEVEONLY	**95.4%**	0.4	**95.5%**	0.7	**70.3%**	0.9
DELETEONLY	85.7%	7.5	83.0%	9.0	45.6%	24.6
DELETEANDRETRIEVE	88.7%	8.4	**96.8%**	7.3	48.0%	22.8

Table 4: Automatic evaluation results. "Classifier" shows the percentage of sentences labeled as the target attribute by the classifier. BLEU measures content similarity between the output and the human reference.

Prabhumoye et al. Style Transfer Through Back-Translation ACL2018

Main ideas:

1. Using MT to get the content.

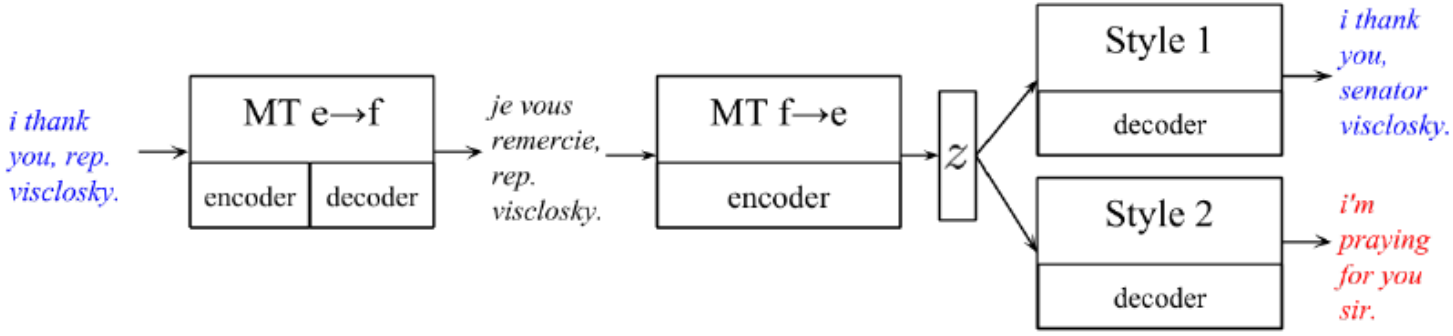


Figure 1: Style transfer pipeline: to rephrase a sentence and reduce its stylistic characteristics, the sentence is back-translated. Then, separate style-specific generators are used for style transfer.

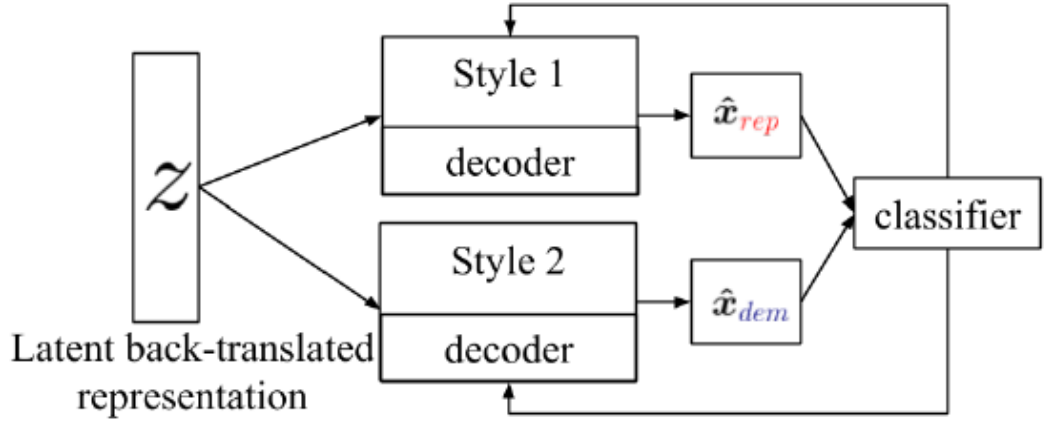


Figure 2: The latent representation from back-translation and the style classifier feedback are used to guide the style-specific generators.

$$\min_{\theta_{gen}} \mathcal{L}_{gen} = \mathcal{L}_{recon} + \lambda_c \mathcal{L}_{class}$$

Style Transfer as Unsupervised Machine Translation

Model

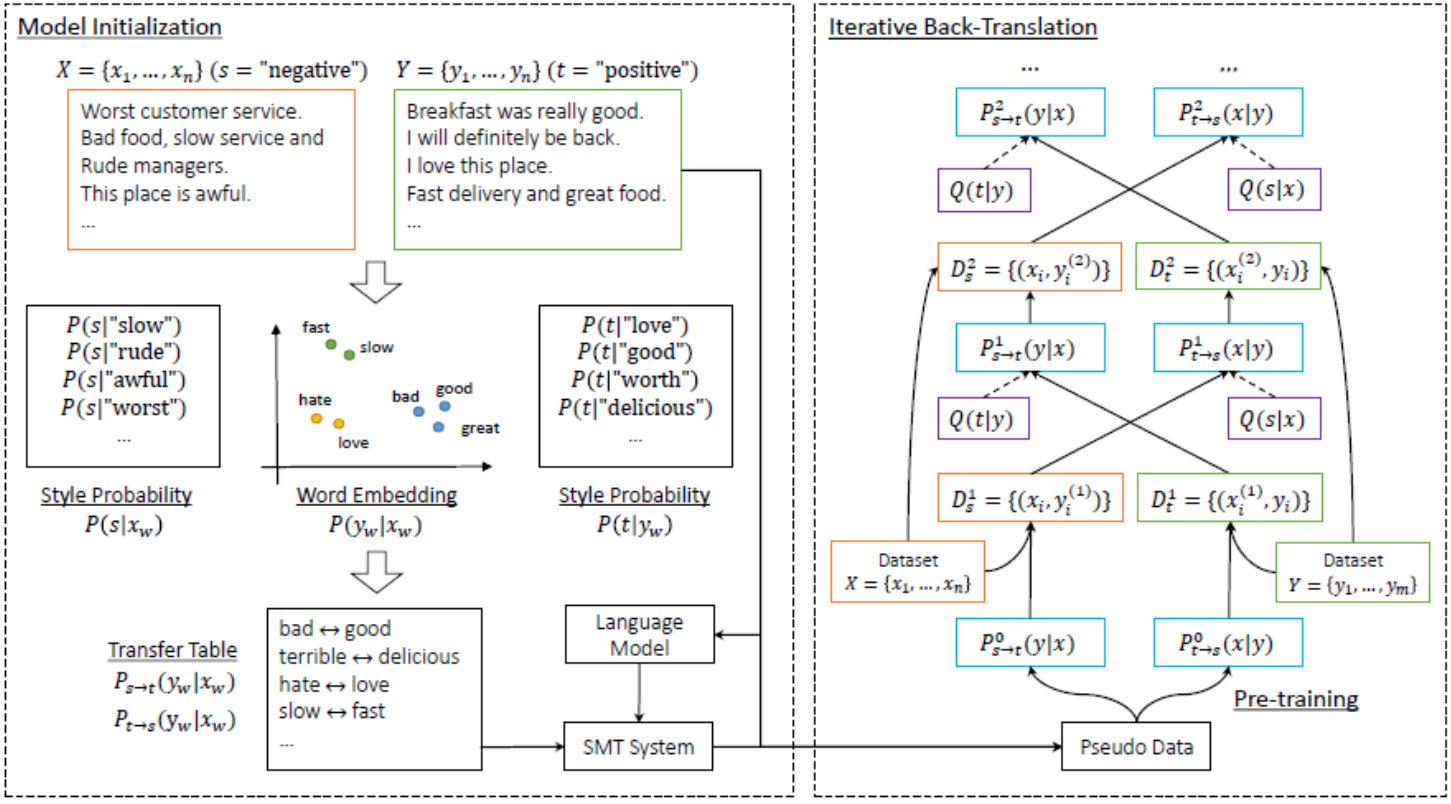


Figure 2: Illustration of the overall training framework of our approach. This framework consists of model initialization and iterative back-translation components, in which $P(s|x_w)$ and $P(t|y_w)$ denote style preference probabilities of words x_w and y_w, $P(y_w|x_w)$ represents word similarity defined in the embedding space, $P_{s\to t}(y_w|x_w)$ and $P_{t\to s}(x_w|y_w)$ stand for the transfer probability of different words, $P_{s\to t}(y|x)$ and $P_{t\to s}(x|y)$ are source-to-target and target-to-source style transfer models, $Q(s|x)$ and $Q(t|y)$ denote the probabilities that a sentence belongs to different styles, and they are used to punish poor pseudo sentence pairs with wrong attributes.

Method

Construct vocabulary

1.学些去词级别的转换知识。这里用三部分概率来模拟概率：1）两个风格概率。这是统计信息，描述一些词与属性的关系（概率)。2）词向量对齐概率。这个是通过比较词向量相似度得到的。

$$
\begin{aligned}
P_{s\to t}(y_w|x_w) = P(y_w|x_w,s,t) &= \frac{P(y_w,s,t|x_w)}{P(s,t)} \\
&= \frac{P(s|x_w)P(y_w|x_w,s)P(t|x_w,y_w,s)}{P(s,t)} \\
&\propto P(s|x_w)P(y_w|x_w)P(t|y_w)
\end{aligned}
$$

Method

SMT

1.训练两个基于4-gram的语言模型。通过结合词级别的转换概率、语言模型构建一个SMT，以此来获得初始的伪数据对。

Method

Iterative Back-Translation

1. Using current pseudo-parallel data to train NMT
2. Using pre-trained style classifer to punish wrong sentences generated by NMT
3. Sampling output of NMT as new pseudo data

	Yelp		Amazon		Captions	
	Classifier	BLEU	Classifier	BLEU	Classifier	BLEU
CrossAligned	73.2%	9.06	71.4%	1.90	79.1%	1.82
MultiDecoder	47.0%	14.54	66.4%	9.07	66.8%	6.64
StyleEmbedding	7.6%	21.06	40.3%	15.05	54.3%	8.80
TemplateBased	80.3%	22.62	66.4%	33.57	87.8%	**19.18**
Del-Retr-Gen	89.8%	16.00	50.4%	29.27	95.8%	11.98
Our Approach	**96.6%**	**22.79**	**84.1%**	**33.90**	**99.5%**	12.69

Table 3: Automatic evaluation results on Yelp, Amazon and Captions datasets. "Classifier" shows the accuracy of sentences labeled by the pre-trained style classifier. "BLEU(%)" measures content similarity between the output and the human reference.

	Yelp				Amazon				Captions			
	Att	Con	Gra	Suc	Att	Con	Gra	Suc	Att	Con	Gra	Suc
CrossAligned	3.1	2.7	3.2	10%	2.4	1.8	3.4	6%	3.0	2.2	3.7	14%
MultiDecoder	2.4	3.1	3.2	8%	2.4	2.3	3.2	7%	2.8	3.0	3.4	16%
StyleEmbedding	1.9	3.5	3.3	7%	2.2	2.9	3.4	10%	2.7	3.2	3.3	16%
TemplateBased	2.9	3.6	3.1	17%	2.1	3.5	3.2	14%	3.3	**3.8**	3.3	23%
Del-Retr-Gen	3.2	3.3	3.4	23%	2.7	**3.7**	3.8	22%	3.5	3.4	**3.8**	32%
Our Approach	**3.5**	**3.7**	**3.6**	**33%**	**3.3**	**3.7**	**3.9**	**30%**	**3.6**	**3.8**	3.7	**37%**

Table 4: Human evaluation results on Yelp, Amazon and Captions datasets. We show average human ratings for style transfer accuracy (Att), preservation of meaning (Con), fluency of sentences (Gra) on a 1 to 5 Likert scale. "Suc" denotes the overall success rate. We consider a generated output "successful" if it is rated 4 or 5 on all three criteria (Att, Con, Gra).