A Comparative Study on Disentangled Style Representation: from Sentiment to Formality

Yonglin Wang
Brandeis University
yonglinw@brandeis.edu

Xiaoyu Lu
Brandeis University
loewilu@brandeis.edu

Abstract

In this project, we empirically examined a technique for text formality extraction, namely, disentangled representation learning. We discussed the model architecture in the original work, whose datasets were on sentiment transfer, in the light of our current style transfer task on formality. Our results proved that, since the linguistic assumption and statistical approximation of style and content in the original sentiment transfer tasks could not generalize well to formality transfer tasks, the performance of the original system on our formality transfer dataset was not as good as it was compared to the original application to sentiment transfer datasets.

1 Introduction

There has been growing body of research in text style transfer aiming to train models that modify specific attributes of input text (e.g., sentiment or formality) while preserving the remaining content (Riley et al., 2020) . A sub-branch of research is dedicated to disentangling the style information in the latent space and in corporate such information in the decoding process (Fu et al., 2018) (John et al., 2019).

With an emphasis on style encoding, we will specifically examine the techniques used in the paper of interest (John et al., 2019) to enforce the latent space of the VAE model to specifically encode style information, disentangled from the content. It is also worth noting that although their approach requires labeled data, the dataset does not need to be parallel. While their work focused on the transfer of sentiment (between positive and negative), in this project, we will empirically examine the effectiveness of their strategy on a different style transfer corpus—Grammarly's Yahoo Answers Formality Corpus (Rao and Tetreault, 2018).

2 Original Work

Since this project primarily focus on *encoding* text styles, we consider metrics related to transfer out of scope for our discussion. Instead, the discussion in this section will focus on the type of architecture and model design that enabled (John et al., 2019) to effectively encode style information in the latent space, while disentangling it from the content information.

2.1 Model Architecture and Loss Functions

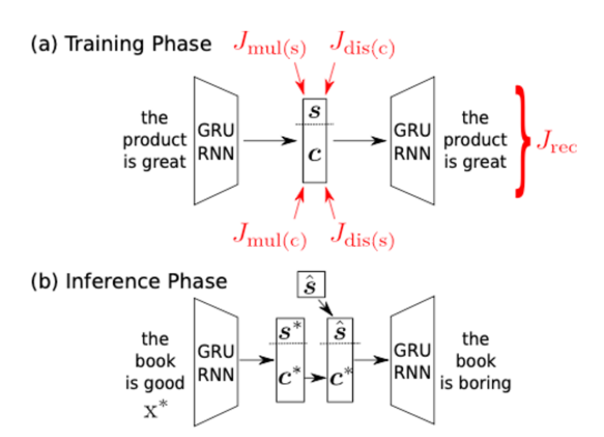

Figure 1: Overview of approach in (John et al., 2019). Note that the 136-dimensional latent vector in (a) is considered as a concatenation of the style (**s**, 8-dimensional) and content (**c**, 128-dimensional) vectors.

As shown in Figure 1, the models being optimized during training are an RNN-VAE, which contains an encoder with parameters $\boldsymbol{\theta}_E$ and a decoder with $\boldsymbol{\theta}_D$, two adversarial classifiers with $\boldsymbol{\theta}_{dis(c)}$ and $\boldsymbol{\theta}_{dis(s)}$, and two multi-task classifiers with $\boldsymbol{\theta}_{mul(c)}$ and $\boldsymbol{\theta}_{mul(s)}$.

During training, the two sets of adversarial discriminator parameters, $\boldsymbol{\theta}_{dis(c)}$ and $\boldsymbol{\theta}_{dis(s)}$, were updated first, followed by the update of the rest of the sets of parameters. During inference time, only the

RNN-VAE model will be involved in generating the style-transferred sentence.

The reconstruction loss of the VAE, $J_{AE}(\boldsymbol{\theta}_E, \boldsymbol{\theta}_D)$, is combined with 4 additional accessory losses, multi-task loss and adversarial loss for each of style and content. The loss functions are used to maximally disentangle the information encoded by the two parts of the hidden vector—one part for content (**c**), the other for style (**s**).

Multi-task loss was incorporated to maximize the information we would like to encode in the latent space; on the other hand, adversarial loss minimized the predictability of information we would like to exclude from the latent space. The following subsections will present the design of these loss functions in detail.

2.2 Multi-task loss

Multi-task losses in this work are defined as the cross-entropy loss of the multitask classifier. The multi-task loss is defined such that the multiple tasks involved during training are 1) to encode the sentence correctly and 2) to make correct prediction based on the disentangled vector. Therefore, during training, both the encoder parameters $\boldsymbol{\theta}_E$ and multi-task classifier parameters will be optimized.

More specifically, in a style-oriented multi-task loss function $J_{mul(s)}(\boldsymbol{\theta}_E, \boldsymbol{\theta}_{mul(s)})$, the classifier's task is to correctly predict the style label based on the style vector.

The design of the content-oriented multi-task loss function $J_{mul(c)}(\boldsymbol{\theta}_E, \boldsymbol{\theta}_{mul(c)})$ is based on a proposed assumption that "content" can be approximated using a bag-of-words (BoW) model. Given a latent content vector **c**, the output of the classifier is a softmax probability vector with length equal to the size of the vocabulary, with each element indicating the probability of a specific word in the vocabulary occurring in the input sentence. A good classifier in this case should assign high probability to the words that actually appear in the input sentence than the absent ones in the vocabulary.

2.3 Adversarial loss

Adversarial losses in this work are defined as the maximal entropy of the discriminator. In general, we swap the input vectors for the two types of classifier design described above, i.e. now classifying style label given the content vector and classifying BoW cooccurrence probability given the style vector, and use the classifiers in this case as adversarial discriminators.

A style-oriented loss function $J_{adv(s)}(\boldsymbol{\theta}_E)$ was implemented to deliberately discriminate the true style label using the content vector **c**—since we do not wish content to contain any distinctly style-related information, we will punish the model if the adversarial logistic classifier predicts the style label correctly based on the content vector **c** alone. In this way, the encoder is trained to produce a content space that does not contain style information.

Along the same line of argument, to ensure that content information is not contained in the style vector **s**, adversarial loss $J_{adv(c)}(\boldsymbol{\theta}_E)$ was also applied using an adversarial softmax classifier that attempts to predict BoW features from the style vector **s** alone.

2.4 Overall Loss

$$
\begin{aligned}
&J_{ovr} = J_{AE}(\boldsymbol{\theta}_E, \boldsymbol{\theta}_D) \\
&+ \lambda_{mul(s)} J_{mul(s)}(\boldsymbol{\theta}_E, \boldsymbol{\theta}_{mul(s)}) - \lambda_{adv(s)} J_{adv(s)}(\boldsymbol{\theta}_E) \\
&+ \lambda_{mul(c)} J_{mul(c)}(\boldsymbol{\theta}_E, \boldsymbol{\theta}_{mul(c)}) - \lambda_{adv(c)} J_{adv(c)}(\boldsymbol{\theta}_E)
\end{aligned} \tag{1}
$$

During training, the overall loss consists of all the terms mentioned above are combined as shown in Equation 1 , where $\lambda_{mul(s)}$, $\lambda_{adv(s)}$, $\lambda_{mul(c)}$, and $\lambda_{adv(c)}$ are all tunable hyperparameters.

As a result of the combined loss function, the style vectors after training in their experiment are clearly disentangled in the latent space, while the content vectors are entangled (Figure 2).

Moreover, to analyze how sentiment and content latent vectors were disentangled, they used classification accuracy of three style predictors: the adversarial style classifier with $\boldsymbol{\theta}_{dis(c)}$, whose input in the content vector (**c**), the multitask style classifier with $\boldsymbol{\theta}_{mul(s)}$, whose input is the style vector (**s**), and a separate style classifier (not involved in the style transfer system) trained on the entire latent space vector ([**s**;**c**]). In Table 1, the classification accuracies suggest that the information encoded by style vectors are helpful in predicting true labels, while content vectors alone are not.

3 Dataset

We conducted our experiments on the GYAFC corpus (Rao and Tetreault, 2018). Table 2 and Table 3 show the example input sentences fed to the model, from the formality corpus and the sentiment datasets used in (John et al., 2019). Due to considerations of vocabulary size and fair comparison, we also lowercased all of our formality data and

Latent Space	None (Majority Guess)	Content Space	Style Space	Style and Content
Yelp	0.602	0.697	0.974	0.974
Amazon	0.512	0.693	0.810	0.810

Table 1: Classification accuracy on latent spaces on two sentiment corpora.

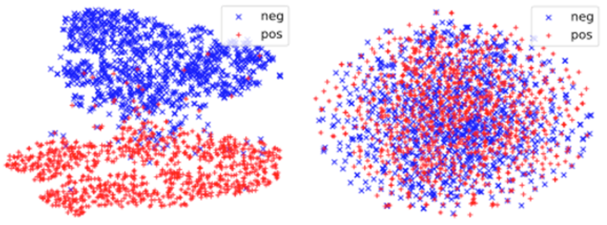

Figure 2: t-SNE plots of the disentangled style and content spaces in the original system by (John et al., 2019) on Yelp dataset.

removed punctuation, although these are in fact also very important aspects of formality (cf. differences between input sentences before and after preprocessing in Table 2).

Even with casing and punctuation information removed, there are still crucial ways in which the formality data has more variation than the sentiment analysis data. Given the input and the satisfactory output in the sentiment transfer task in Table 3, it seemed to be enough to only consider lexically substituting certain sentiment words to their opposite to successfully complete the task. This may be one of the motivation for (John et al., 2019) to choose a bag-of-word model to adversarially disentangle content and style spaces.

In contrast, lexical choice is only one of the many factors that affects formality. Since the original GYAFC dataset was a parallel corpus, the possible forms of desired output for each sentence are also listed. As we can see, other factors may include misspelling, elongation, syntax, context-dependent details.

We will discuss the effect of these crucial differences in properties on modeling later in Section 5.

4 Goals and Hypotheses

The goal of this project is to determine if the results pertinent to style encoding in the original work can be replicated. However, we do not expect the proposed method to work as well for our task as it does on the sentiment transfer task. The main reason is that the original paper naively assumes that different styles entail only different choice of sentiment-related words and that other style-neutral, i.e. sentiment-neutral, words would remain unchanged. This assumption leads to their choices of 1) lowercasing and stripping punctuations during preprocessing, 2) using BoW features to approximate content information, and 3) excluding "style-related" words such as sentiment words in the BoW features.

Comparing this naïve linguistic assumption about style with the aforementioned linguistic properties of formality in Section 3, we can see that the approach in their work may not be able to properly represent style and content in formality encoding and transfer. In this regard, we suspect that the original system would not be able to have the same promising performance in our formality encoding task.

Specifically, we expect to see 1) significantly less disentanglement between formal and informal styles vectors in the t-SNE plot, 2) no change or less entanglement in the t-SNE plot of the content space, 3) lower formality classification accuracies across the board during inference.

5 Experiment and Result Analysis

As shown in Table 4, we experimented with two sets of hyperparameter settings. One set used all of the original default. To see if reducing the emphasis on content can alleviate the untenable approximation of content information, in the other set (referred to as "custom" thereafter), we lowered the weights of the content-related losses in the overall loss function.

Table 5 shows the result of the three types of aforementioned classifiers. First, we can see that there is disentanglement in that incorporating style space information in formality style prediction achieved better accuracy than using content vector alone. Second, it is worth noting that the difference in accuracy is not as drastic as in the original work, suggesting that the approach is less effective for formality prediction. Lastly, the custom setting performs overall worse than the default setting. This suggests that lowering content-oriented losses alone may not be enough for adjusting the system to encoding formality style information properly.

	Input sentence before preprocessing	**Input sentence after preprocessing**	**Reference output (from parallel corpus)**
Formal → Informal	I very much enjoy this song.	i very much enjoy this song	I LOOOOOVVVVVVVEEE this song SOOO Much!!!!!!
	What exactly are you stating?	what exactly are you stating	what the hell do u mean???
Informal → Formal	Beyonce can't sing, dance, or act and Rihanna(who?)	beyonce can't sing dance or act and rihanna who	I do not think Beyonce can sing, dance, or act. You mentioned Rihanna, who is that?
	Hey check out this website they list alot of TV sitcoms on DVD	hey check out this website they list alot of tv sitcoms on dvd	Visit this website, it lists many television situational comedies which are available on DVD.

Table 2: Examples from the GYAFC corpus

	Input sentence after preprocessing	**Acceptable Transferred Sample**
Pos → Neg	the waitresses are friendly and helpful	the waitresses are rude and are lazy
	the restaurant itself is romantic and quiet	the restaurant itself was dirty
Neg → Pos	the interior is old and generally falling apart	the interior is old and noble
	they are clueless	they are genuinely professionals

Table 3: Examples from the review corpora after preprocessing, and their respective sentiment-transferred output.

	Style-oriented		**Content-oriented**	
Type of Loss	**Multitask** ($\lambda_{mul(s)}$)	**Adversary** ($\lambda_{adv(s)}$)	**Multitask** ($\lambda_{mul(c)}$)	**Adversary** ($\lambda_{adv(c)}$)
Default	10	1	3	0.03
Custom	10	1	1	0.01

Table 4: Loss function weight hyperparameters in our experiments.

Examples of generated sentences can be found in Supplemental Material A.

The plots in Figure 3 and 4 give a qualitative representation of the relative distribution of the style and content vectors. We can see that the sentences do not distinguish in content space as intended by the content-oriented loss functions. While the style vectors still separate, we can see that the separation is not as clear as in the original sentiment dataset.

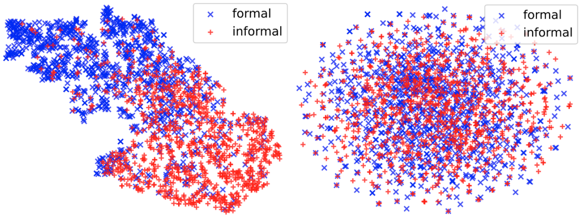

Figure 3: t-SNE plots of the disentangled spaces in default setting. Left: style; right: content.

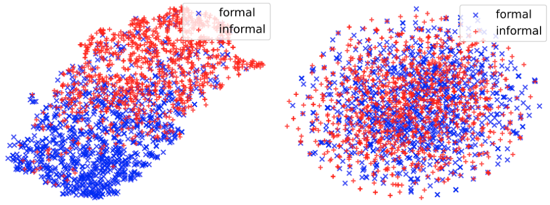

Figure 4: t-SNE plots of the disentangled spaces in custom setting. Left: style; right: content.

6 Conclusions and Discussion

It is clear from the results that the original sentiment style transfer approach described in (John et al., 2019) do not generalize well to formality transfer tasks, especially in terms of encoding formality style. In addition to the reasons mentioned in Section 4, this section will briefly discuss other possible explanations and fixes for the underperforming issue.

First, we may consider some modification to the naïve BoW representation of content information, such as the incorporation of byte-pair-encoding or a curated formality-specific vocabulary set to be excluded from the BoW features. See Supplemental Material B for a list of candidates for curating formality-specific vocabulary.

Second, due to time constraint, there were many different hyperparameters that were left unexplored. For example, we can try using grid search to find better weights for the loss function.

Moreover, the CNN classifier for adversarial training may need to be fine-tuned as well, or we may consider implementing a different classifier model, such as fastText classifier (Joulin et al., 2017), which has a best accuracy of 0.91 in predicting the binary formality labels of raw sentences.

References

Zhenxin Fu, Xiaoye Tan, Nanyun Peng, Dongyan Zhao, and Rui Yan. 2018. Style transfer in text: Exploration and evaluation. In *AAAI*.

Vineet John, Lili Mou, Hareesh Bahuleyan, and Olga Vechtomova. 2019. Disentangled representation learning for non-parallel text style transfer. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 424–434, Florence, Italy. Association for Computational Linguistics.

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, Valencia, Spain. Association for Computational Linguistics.

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.

Parker Riley, Noah Constant, Mandy Guo, Girish Kumar, David Uthus, and Zarana Parekh. 2020. Textsettr: Label-free text style extraction and tunable targeted restyling.

A Supplemental Material

A.1 Generated Sentences

It is apparent from the examples (Table 6 and Table 7) that the systems do not produce comprehensible style-transferred sentences for our formality transfer task. This is expected due to the low classifier accuracy and low disentanglement in latent space.

A.2 Style-specific Lexicon

The word lists on Table 8 suggest that it is possible to curate a formality-specific vocabulary that will be excluded in the BoW features for content approximation.

Latent Space	None (Majority Guess)	Content Space	Style Space	Style and Content
basic	0.50000000	0.66492147	0.71370776	0.71156592
custom	0.50000000	0.64040933	0.65016659	0.6473108

Table 5: Classifier accuracy in our experiments.

	Actual Sentence	Generated Sentence
Formal → Informal	can you imagine starting a family with him	imagine what you are doing
	i suggest avoiding hot dogs and not watching this movie with your little sister	i watch the little mermaid and i like her sister
Informal → Formal	i like thick because i feel cheated with a thin one	i feel like that i prefer brunettes
	im not a little scary so i can do all the stuff that they do ya know	i am a little girl but i do not like the little stuff but

Table 6: Generated sentences from the default system.

	Actual Sentence	Generated Sentence
Formal → Informal	can you imagine starting a family with him	can you imagine your family and her
	i suggest avoiding hot dogs and not watching this movie with your little sister	i suggest you watch the movie and i think she is a little sister
Informal → Formal	i like thick because i feel cheated with a thin one	i like the size of the size of the size
	im not a little scary so i can do all the stuff that they do ya know	i am not a fan of them but i have not seen it but

Table 7: Generated sentences from the custom system.

Set	Words (separated by whitespace)
Unique to informal (50)	maybe wanna wants wont hell thats thought alot gonna stupid definately i've little doesnt real pretty likes can't cool stuff start that's depends wait kind shes gotta hard hate isnt play guys i'll he's kids yahoo happy cause looks whats guess dude probably sorry dont come yeah says kinda cant
Unique to formal (50)	unable choose received acceptable luck wish difficult years website amusing boyfriend enjoy woman likely understand unsure purchase reason intercourse correct situation matter fond true attempt children interested attractive mother favorite simply able television search similar opinion large appears enjoyable relationship unfaithful aware agree date sexual homosexual band prefer information believe
Found in both labels (50)	best hope help work women mean looking find song talk tell friend think right great time thing love heard person care movies sure girl answer need music want nice going life friends said movie people watch things question look better wrong married like know songs feel girls funny long good

Table 8: 100 most frequent words in each label. Stop words are excluded before ranking.