An Evaluation of Disentangled Representation Learning for Texts

Krishnapriya Vishnubhotla[1,2], **Graeme Hirst**[1], and **Frank Rudzicz**[1,2,3]

[1]Department of Computer Science, University of Toronto
[2]Vector Institute for Artificial Intelligence
[3]Unity Health Toronto
{vkpriya,gh,frank}@cs.toronto.edu

Abstract

Learning disentangled representations of texts, which encode information pertaining to different aspects of the text in separate representations, is an active area of research in NLP for controllable and interpretable text generation. These methods have, for the most part, been developed in the context of text style transfer, but are limited in their evaluation. In this work, we look at the motivation behind learning disentangled representations of content and style for texts and at the potential use-cases when compared to end-to-end methods. We then propose evaluation metrics that correspond to these use-cases. We conduct a systematic investigation of previously proposed loss functions for such models and we evaluate them on a highly-structured and synthetic natural language dataset that is well-suited for the task of disentangled representation learning, as well as two other parallel style transfer datasets. Our results demonstrate that current models still require considerable amounts of supervision in order to achieve good performance.

1 Introduction

The similarity of texts can be assessed along multiple dimensions. They could contain the same topics, as identified by semantic similarity. They could belong to the same genre or be written by the same author, in which case we might identify stylistic similarity. Texts that present a positive sentiment may be considered similar to one another when compared to those that express a negative sentiment, even if they talk about different topics. The similarity of texts, therefore, must be defined together with a frame of reference or a pre-specified dimension of variation.

Text representations obtained by current representation learning methods combine all of these different aspects of a text into a single vector embedding (Conneau et al., 2017; Reimers and Gurevych, 2019). This results in only a fuzzy measure of text similarity when it is calculated using methods such as the cosine distance between vector embeddings. Recently, some research in NLP has focused on learning *disentangled* representations for texts, which aim to capture the different dimensions of variation of a text in separate vector embeddings. These methods have been investigated for style transfer to obtain disentangled representations of content and style (John et al., 2019; Romanov et al., 2019; Cheng et al., 2020), and paraphrase generation for disentangling syntax and semantics (Chen et al., 2019; Balasubramanian et al., 2020). Inspired by parallel developments on style transfer and disentanglement in computer vision, many of them operate within the variational autoencoder framework, where the autoencoder is modified to now encode a text into two latent vectors: one capturing the style (the aspect of variation), and the other capturing the content. Style transfer is then achieved by combining the content vector of the input with a style vector of the target style.

Disentanglement-based models offer two main advantages when compared to end-to-end style transfer methods:

1. Sampling from the latent space of the style embeddings allows for more diverse and controlled stylistic generation.

2. Similarity of documents can now be calculated for each aspect of variation, allowing for finer-grained retrieval.

In this work, we focus on models that aim to disentangle **content** from **form**, or meaning from style, for texts. Thus, style transfer is viewed as a form of paraphrasing, where the paraphrase demonstrates certain stylistic properties. It is important to make this distinction between what constitutes style versus meaning for a text, more so when for-

Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, pages 1939–1951

Meaning Representation	name[nameVariable], food[Indian], customerRating[average]
EXTROVERT	nameVariable is an Indian place, also nameVariable has an average rating, you know.
UNCONSCIENTIOUSNESS	Yeah, mmhm... I don't know. nameVariable is an Indian place with a damn average rating.
CONSCIENTIOUSNESS	Did you say nameVariable? I see, well it is an Indian restaurant with an average rating.
DISAGREEABLE	Actually, basically, everybody knows that nameVariable is an Indian restaurant, also it has an average rating.
AGREEABLE	Let's see what we can find on nameVariable. Well, right, it is an Indian restaurant with a quite average rating.

Table 1: The same meaning representation mapped to different stylistic surface realisations in the PersonageNLG dataset.

mulating style transfer problems, in order to have measurable definitions of what information may and may not be changed by the model. Parallel paraphrase datasets, therefore, are a much-needed resource for the effective evaluation of these models. However, few works on disentangled representation learning actually evaluate their models on such datasets, testing instead only on the non-parallel datasets used for training. Further, some works evaluate exclusively on metrics from the style transfer task, ignoring the retrieval aspect.

The goal of this study is to conduct a systematic and grounded evaluation of various disentangled representation learning models. We first use, as a testbed for our evaluation strategy, a highly-structured Natural Language Generation dataset, PersonageNLG (Oraby et al., 2018), which maps a meaning representation to a set of stylistically different surface realisations corresponding to five personality types (Table 1). This dataset provides us with textual variation and gold-standard annotations for the two dimensions of interest, content and form. The structured and somewhat synthetic nature of this dataset allows us to systematically investigate the quality of the disentangled representations for metrics of aspect-specific retrieval as well as style transfer.

We then extend our experiments to two other parallel style transfer datasets: the GYAFC formality corpus (Rao and Tetreault, 2018), and the Bible dataset (Carlson et al., 2018). Although parallel, they are not annotated for semantic content as the PersonageNLG dataset is; however, they are arguably more representative of the kinds of data we expect to obtain in the real world. Despite testing our models with loss functions that do not require parallel data, we limit ourselves to such datasets for the ease and consistency of evaluation. Our code is publicly available at `github.com/priya22/drl-nlg-eval`.

2 Background

Works on style transfer in NLP operate with varying definitions of what constitutes style. Many choose to define this as a factor of variation in data that can be manipulated, including aspects such as topic and sentiment. This approach has been contested by others who maintain that the semantic content of a text should not be modified when manipulating style. The latter definition fits with what stylometric analysis and linguistics consider to be the style of a text. Thus, the output of a style transfer system should be a paraphrase of the input text.

2.1 Model Architectures

The models used to achieve style transfer fall into a few broad categories. End-to-end sequence transformation models are inspired by machine translation seq-2-seq models, where the translation is done from style A to style B. These sometimes require parallel data, but methods such as backtranslation circumvent that (Prabhumoye et al., 2018; He et al., 2020). Some others look at this as a controlled text generation problem, where the control is generally a categorical variable indicating the desired stylistic class of the output, and is passed along with the input to a text generation module such as an LSTM (Hu et al., 2017; Ficler and Goldberg, 2017).

The focus of this work is on a third class of models that first learn disentangled latent representations of style and not-style (henceforth referred to as content) for a text, and train a generator that takes both vectors as input. To transform a text A into the style of text B, we extract the content vector of the former, the style vector of the latter, and pass them through the generator. Note that here, the style vectors of each text are not the same categorical variable, but rather a vector embedding that encodes the style-specific properties of the text. One can also obtain a single style vector represen-

tation by averaging the style vectors of all texts belonging to that class, as Fu et al. (2018) did; however, we are more interested in disentangling information at the individual text level rather than in corpus-level indicators.

2.2 Relevant Literature

Disentanglement of latent spaces has been widely studied and very successful in computer vision applications, but less so in NLP. This can be attributed to the vague nature of what actually constitutes style as opposed to content for a text, and uncertainty as to whether they can actually be disentangled at all (Lample et al., 2019; Yamshchikov et al., 2019). However, by using some supervision with respect to these two dimensions, researchers have attempted to obtain representations that *for the most part* encode information relating to only style or only semantics.

Romanov et al. (2019) first proposed obtaining separate embeddings of form and meaning of texts. Starting with an encoder-decoder setup, they added adversarial and motivational losses based on style labels that encourage the form vector to encode information relevant to the label. Their models were evaluated on non-parallel datasets with two types of stylistic variation: diachronic language shift and newspaper titles versus scientific paper titles. In parallel work, John et al. (2019) proposed a disentanglement model that appends additional content-based losses, where content is approximated by a bag-of-words representation of the text. Their approach was applied to sentiment transfer for Yelp and Amazon reviews.

Other work has looked at disentangling syntax from the semantics of a text. Chen et al. (2019) proposed a VAE-based model that used parallel paraphrase corpora; this was also the focus of Bao et al. (2019) and Balasubramanian et al. (2020).

All of these works are very similar in the base model architecture and the kinds of loss functions used to guide disentanglement. In the following sections, we consolidate and propose a broad categorization of these losses that we hope will guide future work in this area. We then evaluate these models on parallel style transfer datasets, with ablation studies on the PersonageNLG dataset.

Note on unsupervised disentanglement: While unsupervised approaches such as the β-VAE have been very successful at disentangling factors of variation in visual data (Higgins et al., 2017), we are still far from achieving such a clean separation of the data generating factors for text. A recent promising approach in this direction was presented by Xu et al. (2020), who use pretrained models along with a novel constraint over the latent space of a VAE to control the sentiment and topic of a text.

3 Methodology

3.1 Autoencoder Model

Following previous literature, our encoder module takes as input a text, and computes latent vector embeddings for each aspect: content and form. The decoder takes as input both vectors, and generates output text. The entire autoencoder model is trained to reconstruct the input text.

Let us denote our content and form encoders by E_c and E_f, the decoder by G, and their model parameters by θ_{E_f}, θ_{E_c} and θ_G respectively. Our base loss can thus be written as:

$$L_{AE} = L_{rec} + \beta L_{reg} \tag{1}$$

where

$$L_{rec}(\theta_{E_c}, \theta_{E_f}, \theta_G) = \mathop{\mathbb{E}}_{x}[-\log\ p_g(x \mid E_f(x), E_c(x))] \tag{2}$$

is the reconstruction loss of the autoencoder given input x, p_g is the decoder distribution, and L_{reg} is an additional regularization term. For a Variational Autoencoder (VAE) model, this is the Kullback-Leibler divergence between the latent posterior distributions q of the encoders and the latent prior $p(z)$:

$$L_{reg}(\theta_E) = D_{KL}(q(z \mid x) \parallel p(z)) \tag{3}$$

An alternative regularization for text autoencoders was proposed by Shen et al. (2020), where the AE loss is augmented with a denoising objective. The input text is perturbed with small amounts of "noise" in the form of word deletions or substitutions; the autoencoder is still trained to reconstruct the original text. Here,

$$L_{reg}(\theta_E, \theta_G) = \mathop{\mathbb{E}}_{(x,\tilde{x})}[-\log\ p_g(x \mid E(\tilde{x}))] \tag{4}$$

where $\tilde{x}$ is the noisy version of the input text x. These denoising autoencoders (DAEs) were shown to be more stable than VAEs for text modeling.

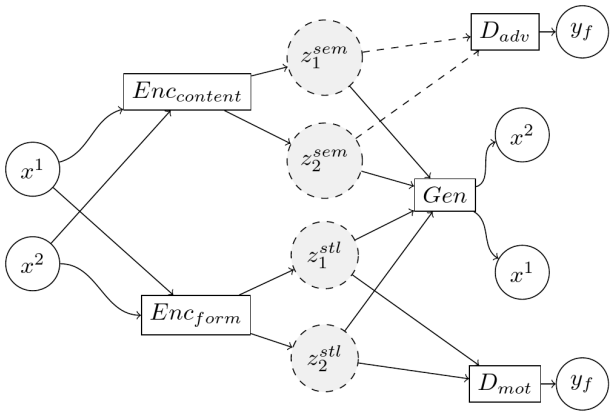

Figure 1: The main components of a Disentangled Representation Learning model. z^{sem} and z^{stl} denote the content and form vectors respectively; each is input to a motivational and an adversarial network. The generator is trained to reconstruct the original input as well as paraphrases.

3.2 Losses for Disentanglement

With our base autoencoder in hand, we can now start adding losses that encourage each latent vector to encode information relevant to the corresponding aspect, i.e, content (semantics) and form (style).

3.3 Proxy-based Losses

Supervised losses are usually based on some form of proxy information present for a specific aspect.

For the form dimension, the most common proxy is class labels that indicate the style of a particular datapoint, such as formal or informal. A stronger proxy could include a list of linguistic attributes of the sentence that are highly indicative of and inform its style. These usually have to be manually defined and extracted, as by John et al. (2019), who use high-polarity sentiment words as a proxy for the sentiment aspect.

An attribute-based proxy for content can be found by looking at the information present in, say, the meaning representation of a sentence (as provided in NLG datasets), or extracting semantics-predictive information such as semantic role labels. John et al. (2019), for example, use the bag-of-words representation of a text as a proxy for semantic information.

These additional losses are usually combined with the autoencoder objective in two ways: as a **motivational** loss, which *encourages* a latent vector to encode the proxy information, and as an **adversarial** loss, which *discourages* a latent vector from encoding the proxy information. Thus, once we define a proxy loss for, say, content, we would append a motivational loss to the content encoder and a corresponding adversarial loss to the form encoder.

Below, we use z_c and z_f to denote the content and form vectors of a text x.

3.3.1 Loss Functions for Form

Motivational: For the datasets that we consider here, and in most real-world applications, we have the stylistic class of a text as a proxy for the form aspect. The motivational and adversarial networks are implemented as classifiers that are trained to predict this label from the corresponding latent representation. The loss function of the former is simply the cross-entropy loss of the classifier:

$$L_{mot}(\theta_D, \theta_{E_f}) = \mathop{\mathbb{E}}_{z_f}[-\log\ D(z_f)] \quad (5)$$

D and θ_D represent the classifier and its parameters respectively.

Adversarial: We now want to ensure that the content vector does not contain any information about the form class of the text. Thus, we aim to *maximize* the entropy of the adversarial classifier. This is the approach followed by many prior works (John et al., 2019; Fu et al., 2018), which we also adopt here, as it can be nicely extended to multi-label classification, which will prove useful in the content-based losses.

Adversarial training occurs in two steps. First, the classifier is trained to predict the form label given the content representation. Then, the content encoder's parameters are updated based on the entropy loss:

$$L_{adv}(\theta_D) = \mathop{\mathbb{E}}_{z_c}[-\log\ D(z_c)] \quad (6)$$

$$L_{adv}(\theta_{E_c}) = \mathop{\mathbb{E}}_{z_c}[\mathbb{H}(D(z_c))] \quad (7)$$

where $\mathbb{H}(D(z_c))$ is the entropy calculated over the classifier-predicted label distribution.

3.3.2 Loss Functions for Content

Proxy information for content is generally rare, and needs to be formulated by means of some heuristic measure. In the case of NLG datasets, we have annotated meaning representations that serve as a good proxy. However, such structured representations of meaning are difficult to obtain for general texts.

Let us assume we have a list of k key-value pairs that represent content, as in the MR from Table 1.

We represent the content proxy as a k-dimensional multi-hot vector y_c, where each dimension y_c^i is a binary indicator of whether key k_i is present in the MR.

Motivational: The motivational loss is thus defined as the multi-label cross-entropy loss over the classifier prediction, similar in form to Eq. 6, but now taking the content vector as input.

Adversarial: In turn, the adversarial content loss is found by first training a multi-label classifier that takes the form vector as input and predicts the content attribute vector, and then training the form encoder to maximize the entropy of this classifier.

3.4 Parallel Losses

These losses require as input a pair of paraphrases, say x^1 and x^2. We obtain the latent vectors for content and form for each of these: $z_c^1, z_f^1, z_c^2, z_f^2$ respectively.

Paraphrase reconstruction loss: Here, we swap the content vectors of the paraphrases, retain the form vectors, and attempt to reconstruct the original inputs. This was used by Chen et al. (2019) to disentangle syntax and semantics in paraphrase corpora.

$$L_{para}(\theta_{E_c}, \theta_{E_f}, \theta_G) = \mathop{\mathbb{E}}_{x_1,x_2}[-\log\ p_g(x^1 \mid z_f^1, z_c^2)] + \mathop{\mathbb{E}}_{x_1,x_2}[-\log\ p_g(x^2 \mid z_f^2, z_c^1)] \quad (8)$$

Distance-based loss: This takes the form of a max-margin loss that aims to keep the cosine similarity between the content embeddings of paraphrases higher than that between a random selection of negative example pairs. This particular loss is used by Chen et al. (2019) and Balasubramanian et al. (2020) to disentangle syntax and semantics, although they differ slightly in the criteria to select positive and negative pairs.

4 Datasets

PersonageNLG Dataset: The PersonageNLG corpus (Oraby et al., 2018) is a set of 88,000 pairs of meaning representations and natural language utterances, based on the E2E challenge dataset. Each utterance is associated with a unique style, which corresponds to one of five personality types: Agreeable, Disagreeable, Conscientious, Unconscientious, and Extrovert. The utterances are obtained by means of a statistical NLG system, and by varying a set of 36 predefined stylistic parameters that specify certain phrase aggregation and pragmatic markers (Table 1). The dataset essentially provides us with a structured and synthetic corpus of textual variation, with each utterance annotated for both content (a meaning representation) and form (the stylistic personality class). This makes it ideal for evaluating the quality of disentangled representations.

GYAFC Dataset: Introduced by Rao and Tetreault (2018), the GYAFC corpus consists of 120,000 parallel sentence pairs that are paraphrased in two styles: formal and informal. See section A.1 for details. GYAFC is one of the very few parallel datasets available for style transfer research in NLP.

Bible dataset: This dataset, compiled by Carlson et al. (2018), consists of eight verse-aligned public domain versions of the Bible; see section A.2 for details. These versions are spread out across different decades, and thus belong to their own unique stylistic class. The natural parallel alignment between verses, as well as the relatively stable nature of their semantic content across time, makes this dataset ideal for studies in style transfer (although surprisingly few works on style transfer use it).

5 Evaluation

The goal of our model is to encode in separate vectors the style-specific and content-specific features of a text. The following metrics guide our similarity measures for content and form:

- **Content (C_{sim}):** For the PersonageNLG dataset, content similarity between two sentences is measured as fraction overlap between content labels (Section 3.3.2). For generated sentences, we use all possible slot values for each field of the Meaning Representation (Table 1) to approximate a bag-of-words content representation, and calculate fraction overlap of content terms in both sentences. For the other two datasets, we use the BLEU scores between the generated text and the target paraphrase as a measure of content preservation.

- **Form (F_{class}, F_{sim}):** For all three datasets, we first train a fasttext[1] classifier on their respective training sets to predict stylistic class given the input text (F_1 scores on the test sets

[1] https://fasttext.cc/

are shown in Table 2). This classifier is then used to predict the style class of a generated text. F_{class} is the F_1 score of the predicted labels for generated texts, using the target labels as ground truth.

Additionally, for the NLG dataset, we use an F_{sim} measure that measures the fraction overlap of non-content words of the two texts, where "non-content" is defined as all words that are not associated with content as defined above.

We divide our evaluation metrics into three groups, based on the capabilities and use-cases of learning disentangled representations.

5.1 Autoencoder Capabilities

Reconstruction: One of the basic functions of our model is as an autoencoder, i.e., a model that can reconstruct the input text from its latent encoding. We use the self-BLEU score between the input (reference) and the generated text to measure reconstruction quality.

5.2 Disentanglement

The quality of disentanglement of representations is assessed in two main ways.

Classification: The first is a classification task that aims to predict the proxy information for each text using the latent vectors. For each of our dimensions of content and form, this gives us four measures corresponding to the accuracy of a classifier trained to predict content (form) information from the content (form) vectors, and that of a classifier trained to predict form (content) information from the content (form) vectors. Ideally, we want the former numbers to be high and the latter to be close to random chance.

Retrieval: As stated, one of the advantages of having disentangled representations for each aspect is that we can now obtain aspect-specific similarity scores. Since all our datasets are parallel paraphrase corpora, we can measure how well the content vectors perform at retrieving paraphrases. For each sentence in our test set, we obtain the cosine similarity scores of its content vector with that of every other sentence, and look at how many of the top-k matches are paraphrases of the input. We evaluate this for $k = 5$ for the GYAFC and Bible datasets, and $k = 1$ for the NLG corpus.

Similarly for form, we find the top-k neighbours for the form vector of each sentence and report the precision@k of retrieving texts from the same stylistic class. This metric is particularly informative for PersonageNLG, where we look at the F_{sim} between the input and the closest match.

Dataset	F_1 **score**
PersonageNLG	0.99
GYAFC	0.87
Bible	0.72

Table 2: Performance of the external fasttext classifier on test sets.

5.3 Style Transfer

Finally, we evaluate the effectiveness of our model for the task of style transfer, by testing with paraphrase pairs. Thus, for each pair of paraphrases in the test set, we obtain the content vector of the first and the form vector of the second, and pass them to the decoder module (and vice-versa). The **content preservation** and **transfer quality** of generated sentences are measured using C_{sim} and F_{class} respectively. We also measure the **fluency** of the generated text by measuring the perplexity of generated sentences with a trigram Kneser-Ney language model trained on the training set of each dataset.

6 Experiments

6.1 Setup

The encoder and decoder of our base model are 2-layer LSTM networks with a hidden size of 64. Both the content and form vectors are of the same size for each dataset: 16 for PersonageNLG and 32 for the others. At each decoder timestep, the concatenated latent vector $z = [z_c, z_f]$ is added to the input to obtain the next prediction. During training, teacher forcing with probability 0.4 is used; we use greedy decoding for the PersonageNLG dataset and and beam search with a beam size of 5 otherwise. Motivational and adversarial classifiers are single-layer linear networks trained with RMSprop.

The GYAFC and NLG datasets come with predefined training and test splits. For the Bible dataset, we use a random stratified split with 65–15–20 split for training, validation, and test respectively.

6.2 Experimental Method

Our goal is to methodologically evaluate the effectiveness of each of these losses for disentangling

content from form. We start with our vanilla autoencoder model (L_{ae}), and at each step, add additional losses based on incorporating some supervised information into our model. The terms we add are guided by some intuition on the kinds of supervision we would expect to see in the real world.

1. **Form losses L_{form}:** This assumes that each text is labeled with a class that indicates its stylistic category, such formal / informal, Shakespearean / modern, positive / negative, etc. This enables us to append two of our losses to the base loss: the motivational and adversarial form losses (Section 3.3.1).

2. **Motivational only L_{mot}:** We now add our proxy information for content. We first keep only the motivational losses and remove the adversarial losses for each aspect.

3. **Combined proxy losses L_{proxy}:** We add adversarial losses for form and content to the model above, giving us our full proxy-loss–based model.

4. **Paraphrase losses:** Finally, we add the parallel losses detailed in Section 3.4, taking advantage of our parallel datasets. The alignment of two paraphrases essentially acts as a proxy for the equivalence of semantic content between two texts. Accordingly, we test the following loss combinations:

    - Parallel losses only (Section 3.4) (L_{para});
    - Parallel losses + form losses from point 1 above (L_{para_f}).

Baseline: We additionally compare the effectiveness of these models when compared to a categorical conditional generation model. Here, the form vector is simply an 8-dimensional encoding of the style class label, rather than derived from the input text. The model is trained using the F_{adv} and C_{mot} losses to ensure the content embedding doesn't encode style information, along with the reconstruction loss L_{rec}.

All of these loss combinations are tested on the PersonageNLG dataset, since it is annotated with proxies of both content and form.

7 Results and Discussion

We experimented with both the VAE and the DAE models for our base architecture, and found that the latter was more stable during training. Training the VAE with multiple latent vectors and additional losses often resulted in the model completely ignoring one of the latent vectors; stable modeling of such architectures is still an active area for text data and is left to future work.

7.1 Disentanglement

We first examine how well our models are able to disentangle information pertaining to form and content into the respective latent vectors. Table 3 reports the performances of each model for the metrics discussed in Section 5.2. For conciseness, we only report cross-aspect classification scores in the Classification column, where a lower number indicates better disentanglement. More detailed results with same-aspect scores are presented in Appendix C.1.

In the absence of parallel data, we see that directly adding supervised losses along each dimension is the most effective strategy of disentangling information. Accordingly, the largest performance drops on cross-aspect classification are achieved with the addition of motivation losses L_{form} and L_{mot} for form and content. Adversarial losses do help the overall performance of the model as demonstrated by the drop in cross-aspect classification metrics, especially in the form domain. The maximal supervision afforded by the paraphrase losses L_{para} demonstrates a significant improvement over the best proxy-based model here, indicating that proxy information is generally not complete enough to capture semantic content. However, the lack of similar supervision along the form dimension is reflected in the higher cross-aspect classification scores across all models.

We show t-SNE plots of the form and content vectors computed by each model in Appendix B. The paraphrase model gives us neat clusters of the content vectors corresponding to the different meaning representations.

However, classification numbers alone don't present the whole picture. Our measures of retrieval quality help to isolate the effects of classifier effectiveness from the goodness of the representations alone. For the NLG dataset in particular, the retrieval scores tell us whether the form vector of a text actually encodes information about the linguistic features informing its style, rather than simply encoding enough to be classified in the right stylistic class. Are sentences with similar

	Autoencoder	**Disentanglement**			
	BLEU	Classification: $F_1 \downarrow$		Retrieval $\uparrow$	
Target $\rightarrow$		Form	Content	Form	Content
Input $\rightarrow$		z_c	z_f	z_f	z_c
L_{ae}*	43.4	0.96	0.73	0.57	0.85
L_{form}	−0.07	−0.67	−0.11	0.13	0.08
L_{mot}	0.01	−0.31	**−0.14**	0.08	0.13
L_{proxy}	−0.05	−0.73	−0.13	**0.13**	**0.14**
L_{para}	0.06	−0.68	−0.03	0.11	0.13
L_{para_f}	−0.03	**−0.75**	−0.10	0.12	0.12
$L_{baseline}$	−0.03	−0.65	–	–	0.09

Table 3: Results on reconstruction and disentanglement quality for the PersonageNLG dataset. The first row reports the absolute metric for the base autoencoder model L_{ae}; subsequent rows report the difference from this base score. The first column reports the self-BLEU score between the reconstructed and input text. For classification, we report the cross-aspect F_1 scores of a classifier trained to predict the target aspect from the input. For retrieval, we report the C_{sim} and F_{sim} scores between the input text and its nearest neighbour in the latent space.

textual stylistic or content features closer to each other in the embedding space when compared to other sentences from the same style/content class? The relatively low delta scores when compared to classification performance indicate that this is not the case. While there are marginal improvements, proxy-based losses don't seem to be informative enough to enforce fine-grained structure in the latent space. Our experiments on style transfer in the next section reinforce this conclusion.

7.2 Style Transfer

We swap the form and content vectors of paraphrases from our test set, and evaluate the generated sentences using the metrics defined in Section 5.3. For the NLG dataset, as before, we use term-overlap measures of the similarity for the content and style terms between the generated text and the target paraphrase (C_{sim} and F_{sim}); results are shown in Table 4. Both of these measures are far from their ideal values of 1.0.

The full proxy model L_{proxy} achieves the best performance across all metrics (sample outputs are shown in Appendix C.2). The paraphrase models tend to perform worse than the baseline, especially on the transfer strength metric, F_{sim}. This points to the form vector not being informative enough, especially when no motivational losses are used. It also indicates that the adversarial losses from the proxy-based models were indeed helpful in disentanglement.

We see similar trends in both disentanglement quality and style transfer for the GYAFC and Bible datasets. The quality of text generated was significantly worse when compared to the NLG dataset, but we are still able to encode the style and content-related information in separate vectors with some success, as evidenced by the retrieval scores.

	$C_{sim} \uparrow$	$F_{sim} \uparrow$	Fluency $\downarrow$
L_{ae}	0.29	0.46	1.11
L_{form}	0.28	0.58	1.08
L_{mot}	0.36	0.48	1.09
L_{proxy}	**0.39**	**0.72**	1.10
L_{para}	0.33	0.45	1.11
L_{para_f}	0.35	0.55	1.09
$L_{baseline}$	0.30	0.60	1.06

Table 4: Evaluation of style transfer on the PersonageNLG dataset. Arrows denote desired direction of change.

7.2.1 Does Disentanglement Help?

Our comparison with the categorical baseline $L_{baseline}$ tells us whether learning disentangled representations indeed provides an advantage for the style transfer task. From Table 4, we see that it does quite well on the C_{sim} metric, but is notably lower than L_{proxy} for F_{sim}. This demonstrates the advantage of having a separate vector representation of the form of a text, as opposed to the stylistic class.

7.3 Discussion

Our experiments all demonstrate that direct supervision along each aspect is crucial for learning good aspect-specific representations. This is the case even for the synthetic PersonageNLG dataset,

		Disentanglement		Style Transfer	
		Clf. ↓	Ret. ↑	C_{sim} ↑	F_{class} ↑
GYAFC	L_{base}	0.43	0.20	1.5	0.50
	L_{para_f}	0.35	0.49	3.6	0.83
Bible	L_{base}	0.64	0.25	1.3	0.11
	L_{para_f}	0.12	0.72	3.4	0.39

Table 5: Results on disentanglement quality and style transfer for the GYAFC and Bible datasets. The Clf. column reports the F_1 score of a classifier trained to predict the stylistic class label from the content vector; Ret. reports the P@5 for retrieving paraphrases using the content vectors.

which is by design constrained to have two separable aspects of variation (meaning and style); this is quite rare in real-world data. Indeed, the best performing style transfer model on this dataset, from Harrison et al. (2019), is a heavily supervised one that conditions a seq-2-seq model with annotations for each type of variation in the surface realisations (i.e., the presence of certain tokens).

In the absence of parallel datasets, proxy information is widely used to encourage disentanglement. However, our results show that such supervision is not sufficient to ensure that the embeddings actually encode the linguistic properties that are characteristic of a text's stylistic class (or meaning). With the retrieval experiments on the NLG dataset, we can see that the F_{sim} scores do not significantly differ between the different models. This indicates the difficulty of learning linguistic properties from class labels alone. This also explains the rather high F_1 scores for content classification from form embeddings.

The poor performance of these models on the style transfer task in particular indicates that the decoder, and hence the reconstruction objective itself, is somewhat lacking. This is reflected in the high classification scores of content information from form vectors, especially for the paraphrase model L_{para}. Additional constraints such as the backtranslation loss (Prabhumoye et al., 2018) go some way towards mitigating this issue. On the style transfer task, the baseline model $L_{baseline}$ shows performance comparable to the disentanglement models. One explanation for their poor performance is the inherent defects of variational models of text, such as the latent space vacancy issue, as demonstrated by other works (Xu et al., 2020; Shen et al., 2020).

For evaluation of such disentangled representations, traditional metrics of style transfer, such as the accuracy of an external classifier, are not the best indicators of disentanglement, nor a good demonstration of the usefulness of such embeddings. Most works on disentangled representations for style transfer do end up using a single, averaged vector embedding to inform the decoder of the desired target style. If the goal of learning disentangled representations is to perform style transfer between two classes, then a conditioned language model such as that of Ficler and Goldberg (2017) would suffice.

A more useful use-case for disentangled representations is for calculating aspect-specific similarity and retrieval between texts. However, it is not clear whether we can achieve such disentanglement with current models without fine-grained supervision along each aspect. While the NLG dataset provides us with the necessary supervision to introduce such constraints (via adversarial losses), and also evaluate them, such supervision is not available for real-world datasets.

8 Conclusion

Encoding the different factors of variation in data in separate embeddings is a desirable goal for learning robust and interpretable text representations, as well as for controllable text generation. While style transfer, and sentiment transfer in particular, has guided most of the prior research in this area, we have shown that the associated metrics and datasets are not entirely representative of the goals of learning disentangled text representations. We re-purposed an existing NLG dataset for this task instead, and performed a stronger evaluation of current models for disentangled representation learning. We have also shown that heavy supervision is needed along each aspect to obtain useful representations. Improvements in variational generative models that can overcome issues of posterior collapse and the use of decoding constraints stronger than the reconstruction loss would greatly benefit such models.

Acknowledgements

We thank the reviewers for their helpful comments. This work was financially supported by Natural Sciences and Engineering Research Council of Canada, and carried out with resources provided by the Vector Institute.

References

Vikash Balasubramanian, Ivan Kobyzev, Hareesh Bahuleyan, Ilya Shapiro, and Olga Vechtomova. 2020. Polarized-VAE: Proximity based disentangled representation learning for text generation. *arXiv preprint arXiv:2004.10809*.

Yu Bao, Hao Zhou, Shujian Huang, Lei Li, Lili Mou, Olga Vechtomova, Xinyu Dai, and Jiajun Chen. 2019. Generating sentences from disentangled syntactic and semantic spaces. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6008–6019.

Keith Carlson, Allen Riddell, and Daniel Rockmore. 2018. Evaluating prose style transfer with the Bible. *Royal Society Open Science*, 5(10):171920.

Mingda Chen, Qingming Tang, Sam Wiseman, and Kevin Gimpel. 2019. A multi-task approach for disentangling syntax and semantics in sentence representations. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2453–2464.

Pengyu Cheng, Martin Renqiang Min, Dinghan Shen, Christopher Malon, Yizhe Zhang, Yitong Li, and Lawrence Carin. 2020. Improving disentangled text representation learning with information-theoretic guidance. *arXiv preprint arXiv:2006.00693*.

Alexis Conneau, Douwe Kiela, Holger Schwenk, Loïc Barrault, and Antoine Bordes. 2017. Supervised learning of universal sentence representations from natural language inference data. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 670–680.

Jessica Ficler and Yoav Goldberg. 2017. Controlling linguistic style aspects in neural language generation. In *Proceedings of the Workshop on Stylistic Variation*, pages 94–104.

Zhenxin Fu, Xiaoye Tan, Nanyun Peng, Dongyan Zhao, and Rui Yan. 2018. Style transfer in text: Exploration and evaluation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

Vrindavan Harrison, Lena Reed, Shereen Oraby, and Marilyn Walker. 2019. Maximizing stylistic control and semantic accuracy in NLG: Personality variation and discourse contrast. In *Proceedings of the 1st Workshop on Discourse Structure in Neural NLG*, pages 1–12.

Junxian He, Xinyi Wang, Graham Neubig, and Taylor Berg-Kirkpatrick. 2020. A probabilistic formulation of unsupervised text style transfer. *arXiv preprint arXiv:2002.03912*.

Irina Higgins, Loïc Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. 2017. beta-VAE: Learning basic visual concepts with a constrained variational framework. In *5th International Conference on Learning Representations, ICLR 2017*.

Zhiting Hu, Zichao Yang, Xiaodan Liang, Ruslan Salakhutdinov, and Eric P Xing. 2017. Toward controlled generation of text. In *International Conference on Machine Learning*, pages 1587–1596.

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.

Guillaume Lample, Sandeep Subramanian, Eric Michael Smith, Ludovic Denoyer, Marc'Aurelio Ranzato, and Y-Lan Boureau. 2019. Multiple-attribute text rewriting. In *7th International Conference on Learning Representations (ICLR 2019)*.

Shereen Oraby, Lena Reed, Shubhangi Tandon, TS Sharath, Stephanie Lukin, and Marilyn Walker. 2018. Controlling personality-based stylistic variation with neural natural language generators. In *Proceedings of the 19th Annual SIGdial Meeting on Discourse and Dialogue*, pages 180–190.

Shrimai Prabhumoye, Yulia Tsvetkov, Ruslan Salakhutdinov, and Alan W Black. 2018. Style transfer through back-translation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 866–876.

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.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3973–3983.

Alexey Romanov, Anna Rumshisky, Anna Rogers, and David Donahue. 2019. Adversarial decomposition of text representation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 815–825.

Tianxiao Shen, Jonas Mueller, Regina Barzilay, and Tommi Jaakkola. 2020. Educating text autoencoders: Latent representation guidance via denoising. In *International Conference on Machine Learning*, pages 8719–8729. PMLR.

Peng Xu, Jackie Chi Kit Cheung, and Yanshuai Cao. 2020. On variational learning of controllable representations for text without supervision. In *International Conference on Machine Learning*, pages 10534–10543. PMLR.

Ivan P Yamshchikov, Viacheslav Shibaev, Aleksander Nagaev, Jürgen Jost, and Alexey Tikhonov. 2019. Decomposing textual information for style transfer. In *Proceedings of the 3rd Workshop on Neural Generation and Translation*, pages 128–137.

Appendices

A Parallel Style Datasets

A.1 GYAFC Corpus

The *Grammarly's Yahoo Answers Formality Corpus*, or GYAFC for short, is a benchmark corpus for formality style transfer in NLP[2]. It consists of a total of 120,000 informal / formal sentence pairs, split into training, validation, and test sets.

Sentences were initially sampled from the Yahoo Answers L6 corpus, and formal and informal rewrites from each were collected from workers on Amazon Mechanical Turk (Rao and Tetreault, 2018). Table 6 shows example paraphrases from this corpus.

A.2 Bible Dataset

More than 30 English translations of the Bible have been published over the course of four centuries, the earliest being the King James Version of 1611. These versions are all highly parallel, aligned by verse, and are high-quality translations due to the importance of the source. Carlson et al. (2018) identified 8 of these versions that are in the public domain and released aligned corpora for each[3]. Table 7 shows a sample verse paraphrased in each of the 8 versions we consider. Each version consists of 31,096 verses, giving us close to 870,000 paraphrase pairs. We first split this into an 80–20 development–test split; the development set is further split into training and validation sets with the same ratio.

Formal	I'd say it is punk though.
Informal	However, I do believe it to be punk.
Informal	Gotta see both sides of the story.
Formal	You have to consider both sides of the story.

Table 6: Sample paraphrases from the GYAFC dataset.

[2]https://github.com/raosudha89/GYAFC-corpus

[3]https://github.com/keithecarlson/StyleTransferBibleData

B t-SNE Visualization

t-Distributed Stochastic Neighbor Embedding is a non-linear dimensionality reduction technique useful for visualizing high-dimensional data. Figure 2 shows t-SNE plots of the form vectors (left column) and content vectors (right column) for sentences in the test set of the PersonageNLG dataset, for each of the loss function combinations we tested. Adding the supervised losses for form successfully groups the form vectors together into five clusters for each of the personality classes. While content vectors also show some clustering with the adversarial and motivational losses, paraphrase losses here are the most effective at grouping them into neat clusters for each of the unique meaning representations in our test set.

C More Results

C.1 Detailed Disentanglement Evaluation

In Table 8, we present a more detailed evaluation on the disentanglement metrics for our models. Here, the Classification column presents both same-aspect and cross-aspect F_1 scores. Higher scores for the former and lower scores for the latter indicate better disentanglement.

We notice that form information is not effectively removed from the content representations, as evidenced by the higher F_{sim} scores for the content vectors z_c. This is a consequence of the weaker label-based proxy used for style, as opposed to the Meaning Representation-based attribute proxy for content.

C.2 Style Transfer Outputs

Table 9 shows sample outputs from the style transfer experiments on PersonageNLG. The model used is the best performing proxy-based model L_{proxy}, with motivational and adversarial losses for both style and content. Two paraphrases with different styles are first encoded into their form and content vectors. The output is generated by passing the form vector of the first sentence and the content vector of the second to the decoder. We see that the model transfers the form attributes quite well across the inputs, but content attributes are not retained perfectly.

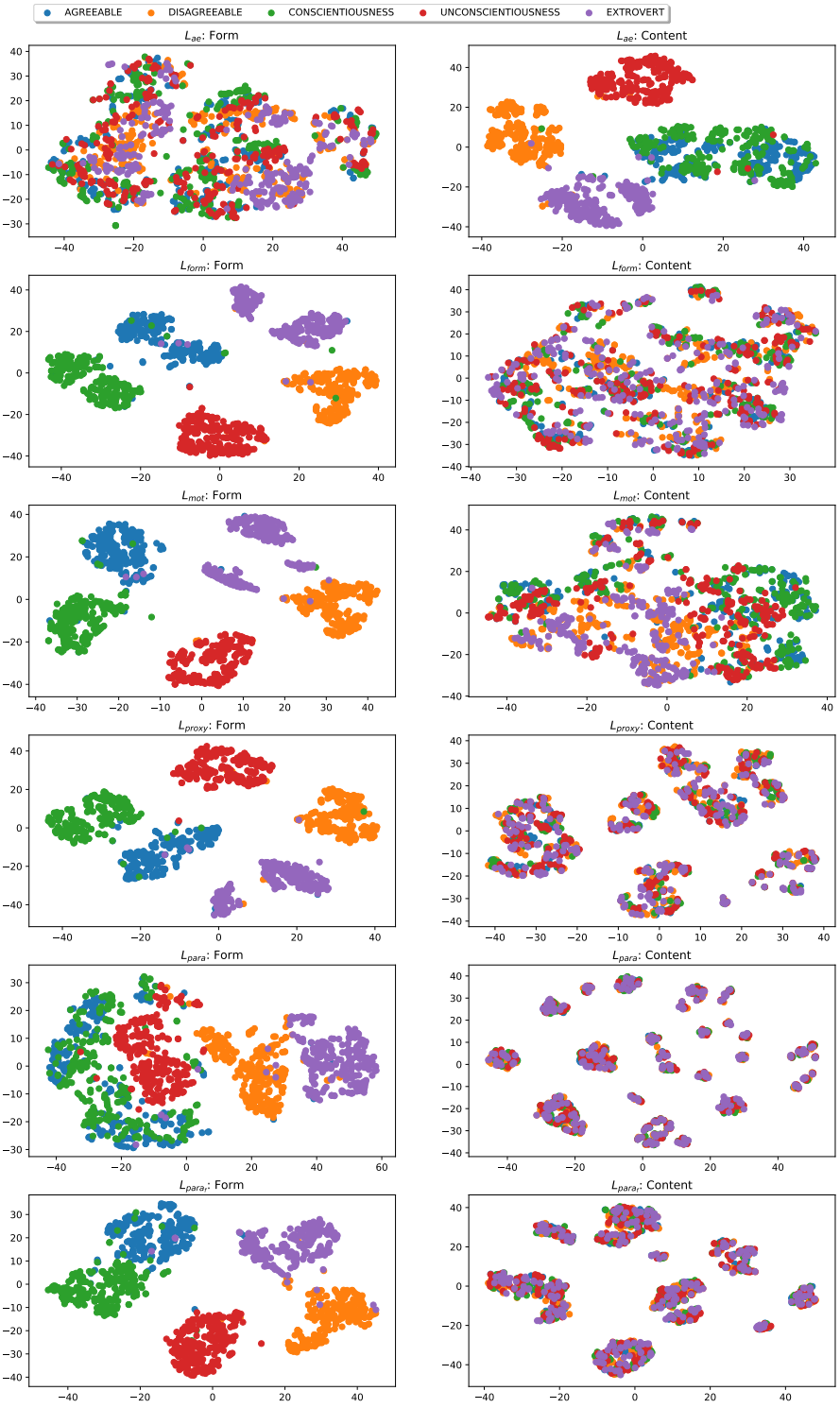

Figure 2: t-SNE visualization of form and content vectors for the PersonageNLG dataset, for each of our models. We see that the paraphrase losses enable a clean clustering of the meaning representations across stylistic variations. The domination of extrovert (purple) in some of the conditions is an artifact of the visualization when points fall in the same place.

Version	Verse
KJV	The heart of the prudent getteth knowledge; and the ear of the wise seeketh knowledge.
ASV	The heart of the prudent getteth knowledge; And the ear of the wise seeketh knowledge.
BBE	The heart of the man of good sense gets knowledge; the ear of the wise is searching for knowledge.
DARBY	The heart of an intelligent getteth knowledge, and the ear of the wise seeketh knowledge.
DRA	A wise heart shall acquire knowledge: and the ear of the wise seeketh instruction.
LEB	An intelligent mind will acquire knowledge, and the ear of the wise will seek knowledge.
WEB	The heart of the discerning gets knowledge. The ear of the wise seeks knowledge.
YLT	The heart of the intelligent getteth knowledge, And the ear of the wise seeketh knowledge.

Table 7: The same verse (Proverbs 18:15) paraphrased in 8 different diachronic versions of the Bible, from the Bible dataset: the King James Version (KJV, 1611), American Standard Version (ASV, 1901), Bible in Basic English (BBE, 1965), Darby Bible (DARBY, 1890), Douay-Rheims edition (DRA, 1899), Lexham English Bible (LEB, 2010), World English Bible (WEB, 2000), and Young's Literal Translation (YLT, 1862).

Model	Classification: F_1				Retrieval			
	Form		Content		Form: F_{sim}		Content: C_{sim}	
	$z_f \uparrow$	$z_c \downarrow$	$z_c \uparrow$	$z_f \downarrow$	$z_f \uparrow$	$z_c \downarrow$	$z_c \uparrow$	$z_f \downarrow$
L_{ae}	0.73	0.96	0.58	0.73	0.57	0.95	0.85	0.70
L_{form}	0.98	0.29	0.62	0.62	0.70	0.90	0.93	0.55
L_{mot}	0.98	0.65	0.92	0.59	0.65	0.90	0.98	0.63
L_{proxy}	0.98	0.23	0.92	0.60	0.70	0.85	0.99	0.54
L_{para}	0.95	0.28	0.80	0.70	0.68	0.93	0.98	0.55
L_{para_f}	0.98	0.21	0.75	0.63	0.69	0.87	0.97	0.54

Table 8: Classification and Retrieval scores that measure the quality of disentanglement of information for each of our models, evaluated on the PersonageNLG dataset

Input (Style A)	nameVariable is near nearVariable pal, nameVariable is a restaurant and it isn't family friendly, also the rating is average, you know!
Target (Style B)	You want to know more about nameVariable? Yeah, it isn't rather family friendly with an average rating, also it is sort of near nearVariable, also it is a restaurant, you see?
Output (Style A → Style B)	You want to know more about nameVariable? Oh it is sort of near nearVariable, also it is a restaurant, also it isn't family friendly, you see
Input	nameVariable is moderately priced, also it's in riverside. It is near nearVariable. It is a pub. it's an Italian restaurant. oh God basically, nameVariable is kid friendly.
Target	Yeah, err... I am not sure. nameVariable is an Italian place near nearVariable in riverside, damn kid friendly and moderately priced and nameVariable is a pub.
Output	Yeah, I am not sure. nameVariable is darn moderately priced in city centre near nearVariable, also it is a coffee shop, also it isn't kid friendly

Table 9: Sample style transfer outputs for the best performing proxy-based model, L_{para}, on the PersonageNLG Dataset.