A Additional Details On the Surrogates

A.1 Proof of Inequality Eq. 6

In this section, we provide a formal proof of the Eq. 6. Let (Z, Y) be an arbitrary pair of RVs with $(Z, Y) \sim p_{ZY}$ according to some underlying pdf, and let $q_{\widehat{Y}|Z}$ be a conditional variational probability distribution on the discrete attributes satisfying $p_{ZY} \ll p_Z \cdot q_{\widehat{Y}|Z}$, i.e., absolutely continuous.

$$I(Z;Y) \geq H(Y) - \text{CE}(\widehat{Y}|Z). \tag{8}$$

Proof: We start by the definition of the MI and use the fact that the maximum entropy distribution is reached for the uniform law in the case of a discrete variable (see (Cover and Thomas, 2006)).

$$\begin{aligned} I(Z;Y) &= H(Y) - H(Y|Z) && (9)\\ &= \text{Const} - H(Y|Z). && (10)\end{aligned}$$

We then need to find the relationship between the cross-entropy and the conditional entropy.

$$\begin{aligned} &\text{KL}(p_{YZ} \| q_{\widehat{Y}Z})\\ =&E_{YZ}\left[\log \frac{p_{Y|Z}(Y|Z)}{q_{\widehat{Y}|Z}(Y|Z)}\right]\\ =&E_{YZ}\big[\log p_{Y|Z}(Y|Z)\big] - E_{YZ}\big[\log q_{\widehat{Y}|Z}(Y|Z)\big]\\ =&-H(Y|Z) + \text{CE}(\widehat{Y}|Z). \end{aligned} \tag{11}$$

We know that $\text{KL}(p_{YZ} \| q_{\widehat{Y}Z}) \geq 0$, thus $\text{CE}(\widehat{Y}|Z) \geq H(Y|Z)$ which gives the result.

The underlying hypothesis made by approximating the MI with an adversarial loss is that the contribution of gradient from $\text{KL}(p_{YZ} \| q_{\widehat{Y}Z})$ to the bound is negligible.

A.2 Proof of Th. 1

Let (Z, Y) be an arbitrary pair of RVs with $(Z, Y) \sim p_{ZY}$ according to some underlying pdf, and let $q_{\widehat{Y}|Z}$ be a conditional variational probability distribution satisfying $p_{ZY} \ll p_Z \cdot q_{\widehat{Y}|Z}$, i.e., absolutely continuous. To obtain an upper bound on the MI we need to upper bound the entropy $H(Y)$ and to lower bound the conditional entropy $H(Y|Z)$.

Upper bound on $H(Y)$. Since the KL divergence is non-negative, we have

$$\begin{aligned} H(Y) &\leq \mathbb{E}_Y\left[-\log q_Y(Y)\right] && (12)\\ &= \mathbb{E}_Y\left[-\log \int q_{\widehat{Y}|Z}(Y|z) p_Z(z) dz\right]. && (13)\end{aligned}$$

Lower bounds on $H(Y|Z)$. We have the following inequalities:

$$\begin{aligned} H(Y|Z) = \mathbb{E}_{YZ}\left[-\log q_{\widehat{Y}|Z}(Y|Z)\right] -\\ \text{KL}(p_{YZ} \| p_Z \cdot q_{\widehat{Y}|Z}), \end{aligned} \tag{14}$$

where $\text{KL}(p_{YZ} \| p_Z \cdot q_{\widehat{Y}|Z})$ denotes the KL divergence. Furthermore, for arbitrary values $\alpha > 1$,

$$\begin{aligned} H(Y|Z) \leq &\mathbb{E}_{YZ}\left[-\log q_{\widehat{Y}|Z}(Y|Z)\right] -\\ &D_\alpha(p_{YZ} \| p_Z \cdot q_{\widehat{Y}|Z}), \end{aligned} \tag{15}$$

where $D_\alpha(p_{YZ} \| p_Z \cdot q_{\widehat{Y}|Z}) =$

$$\frac{1}{\alpha - 1} \log \mathbb{E}_{ZY}\left[R^{\alpha-1}(Z, Y)\right]$$

is the Renyi divergence with

$$R(y, z) = \frac{p_{Y|Z}(y|z)}{q_{\widehat{Y}|Z}(y|z)}.$$

The proof of Eq. 14 is given in Ssec. A.1. In order to show Eq. 15, we remark that Renyi divergence is non-decreasing function $\alpha \mapsto D_\alpha(p_{ZY} \| p_Z \cdot q_{\widehat{Y}|Z})$ in $\alpha \in [0, +\infty)$ (the reader is refereed to (Van Erven and Harremos, 2014) for a detailed proof). Thus, we have $\forall \alpha > 1$,

$$\text{KL}(p_{ZY} \| p_Z \cdot q_{\widehat{Y}|Z}) \leq D_\alpha(p_{ZY} \| p_Z \cdot q_{\widehat{Y}|Z}). \tag{16}$$

Therefore, from expression Eq. 14 we obtain the desired result.

A.3 Optimization of the Surrogates on MI

In this section, we give details to facilitate the practical implementation of our methods.

A.3.1 Computing the entropy $H(Y)$

$$\begin{aligned} H(Y) &\leq \mathbb{E}_Y\left[-\log \int q_{\widehat{Y}|Z}(Y|z) p_Z(z) dz\right]\\ &\approx \mathbb{E}_Y\left[-\log \sum_{i=1}^{n} q_{\widehat{Y}|Z}(Y|z_i)\right] + \text{const.}\\ &\approx -\frac{1}{|\mathcal{Y}|} \sum_{j=1}^{|\mathcal{Y}|} \log \sum_{i=1}^{n} C_{\theta_c}(z_i)_{y_j} + \text{const.} \end{aligned} \tag{17}$$

where $C_{\theta_c}(z_i)_{y_j}$ is the y_j-th component of the normalised output of the classifier C_{θ_c}.

A.3.2 Computing the lower bound on $H(Y|Z)$

The upper bound helds for $\alpha > 1$,

$$H(Y|Z) \approx \mathrm{CE}(Y|Z) - \widehat{D}_\alpha(p_{ZY} \| p_Z \cdot q_{\widehat{Y}|Z}) \approx -\frac{1}{n}\sum_{i=1}^{n} \log q_{\widehat{Y}|Z}(y_i|z_i) - \frac{1}{\alpha-1}\log\sum_{i=1}^{n} R^{\alpha-1}(z_i, y_i). \tag{18}$$

Estimating the density-ratio $R(z, y)$ In what follows we apply the so-called density-ratio trick to our specific setup. Suppose we have a balanced dataset $\{(y_i^p, z_i^p)\} \sim p_{YZ}$ and $\{(y_i^q, z_i^q)\} \sim q_{\widehat{Y}|Z}p_Z$ with $i \in [1, K]$. The density-ratio trick consists in training a classifier C_{θ_R} to distinguish between theses two distribution. Samples coming from p are labelled $u = 1$, samples coming from q are labelled $u = 0$. Thus, we can rewrite $R(z, y)$ as

$$R(z, y) = \frac{p_{Y|Z}(y, z)}{q_{\widehat{Y}|Z}(y, z)} \tag{19}$$

$$= \frac{p_{YZ|U}(y, z|u = 0)}{p_{YZ|U}(y, z|u = 1)} \tag{20}$$

$$= \frac{p_{U|YZ}(u = 0|y, z)}{p_{U|YZ}(u = 1|y, z)} \frac{p_U(u = 1)}{p_U(u = 0)} \tag{21}$$

$$= \frac{p_{U|YZ}(u = 0|y, z)}{p_{U|YZ}(u = 1|y, z)} \tag{22}$$

$$= \frac{p_{U|YZ}(u = 0|y, z)}{1 - p_{U|YZ}(u = 0|y, z)}. \tag{23}$$

Obviously, the true posterior distribution $p_{U|YZ}$ is unknown. However, if C_{θ_R} is well trained, then $p_{U|YZ}(u = 0|y, z) \approx \sigma(C_{\theta_R}(y, z))$, where $\sigma(\cdot)$ denotes the sigmoid function. A detailled procedure for training is given in Algorithm 1.

B Additional Details on the Model

B.1 Baseline Schemas

We report in Fig. 7 the schema of the proposed approach as well as the baselines.

B.2 Architecture Hyerparameters

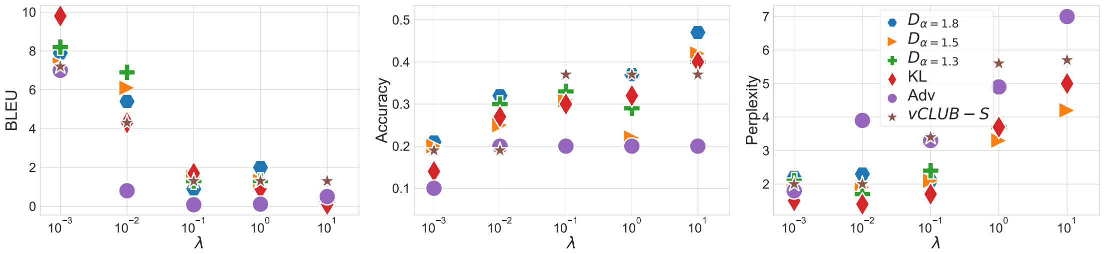
We use an encoder parameterized by a 2-layer bidirectional GRU (Chung et al., 2014) and a 2-layer decoder GRU. Both GRU and our word embedding lookup tables, trained from scratch, and have a dimension of 128 (as already reported by (Garcia et al., 2019), building experiments on higher dimensions produces marginal improvement). The style embedding is set to a dimension of 8. The attribute classifier are MLP and are composed of 3 layer MLP with 128 hidden units and LeakyReLU (Xu et al., 2015) activations, the dropout (Srivastava et al., 2014) rate is set to 0.1. All models are optimised with AdamW (Kingma and Ba, 2014; Loshchilov and Hutter, 2017) with a learning rate of 10^{-3} and the norm is clipped to 1.0. Our model's hyperparameters have been set by a preliminary training on each downstream task: a simple classifier for the fair classification and a vanilla seq2seq (Sutskever et al., 2014; Colombo et al., 2020) for the conditional generation task. The models requested for the classification task are trained during $100k$ steps while 300k steps are used for the generation task.

Algorithm 1 Our method for the fair classification task

```
INPUT: training dataset for the encoder $\mathcal{D}_n = \{(x_1, y_1, l_1), \ldots, (x_n, y_n, l_n)\}$, batch size $m$,
    training dataset for the classifiers and decoder $\mathcal{D}'_n = \{(x'_1, y'_1, l'_1), \ldots, (x'_n, y'_n, l'_n)\}$.
Initialization: parameters $(\theta_e, \theta_R, \theta_c, \theta_d)$ of the encoder $f_{\theta_e}$, classifiers $C_{\theta_R}$, $C_{\theta_c}$, $f_{\theta_d}$
Optimization:
while $(\theta_e, \theta_R, \theta_c, \theta_d)$ not converged do
    for $i \in [1, Unroll]$ do    ▷ Train $C_{\theta_c}$, $C_{\theta_R}$, $f_{\theta_d}$
        Sample a batch $\mathcal{B}'$ from $\mathcal{D}'$
        Update $\theta_R$ based $\mathcal{B}'$ and using $C_{\theta_c}$
        Update $\theta_c$ with $\mathcal{B}'$
        Update $\theta_d$ with $\mathcal{B}'$
    end for
    Sample a batch $\mathcal{B}$ from $\mathcal{D}$    ▷ Train $f_{\theta_e}$
    Update $\theta_e$ with $\mathcal{B}$ using Eq. 3 with $\theta_d$.
end while
OUTPUT: $f_{\theta_e}$, $f_{\theta_d}$
```

C Additional Details on the experimental Setup

In this section, we provide additional details on the metric used for evaluating the different models.



(a) (b) (c)

Figure 4: Numerical experiments on multiclass style transfer using categorical labels. Results include: BLEU (Fig. 4a)); style transfer accuracy (Fig. 4b); sentence fluency (Fig. 4c).

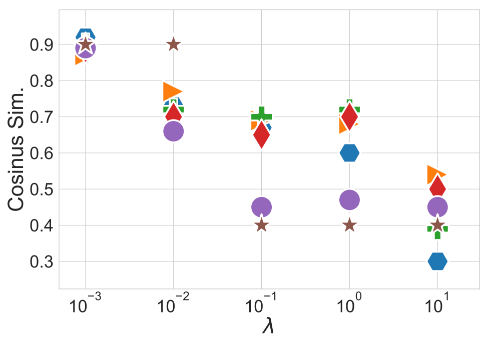


Figure 5: Results of cosine similarity on multiclass style transfer using categorical labels

C.1 Content Preservation: BLEU & Cosine Similarity

Content preservation is an important aspect of both conditional sentence generation and style transfer. We provide here the implementation details regarding the implemented metrics.

BLEU. For computing the BLEU score we choose to use the corpus level method provided in python sacrebleu (Post, 2018) library `https://github.com/mjpost/sacrebleu.git`. It produces the official WMT scores while working with plain text.

Cosine Similarity. For the cosine similarity, we follow the definition of John et al. (2018) by taking the cosinus between source and generated sentence embedding. For computing the embedding we rely on the bag of word model and take the mean pooling of word embedding. We choose to use the pre-trained word vectors provided in `https://fasttext.cc/docs/en/pretrained-vectors.html`. They are trained on Wikipedia using fastText. These vectors in dimension 300 were obtained using the skip-gram model described in Bojanowski et al. (2017); Joulin et al. (2016b) with default parameters.

C.2 Fluency: Perplexity

To evaluate fluency we rely on the perplexity (Jalalzai et al., 2020), we use GPT-2 (Radford et al., 2019) fine-tuned on the training corpus. GPT-2 is pre-trained on the BookCorpus dataset (Zhu et al., 2015) (around 800M words). The model has been taken from the HuggingFace Library (Wolf et al., 2019). Default hyperparameters have been used for the finetuning.

C.3 Style Conservation/Transfer

For style conservation (Colombo et al., 2019) (*e.g.*, polarity, gender or category) we train a fasttext (Bojanowski et al., 2017; Joulin et al., 2016a,b) classifier `https://fasttext.cc/docs/en/supervised-tutorial.html`. We use the validation corpus to select the best model. Preliminary comparisons with deep classifiers (based on either convolutionnal layers or recurrent layers) show that fasttext obtains similar result while being litter and faster.

C.4 Disentanglement

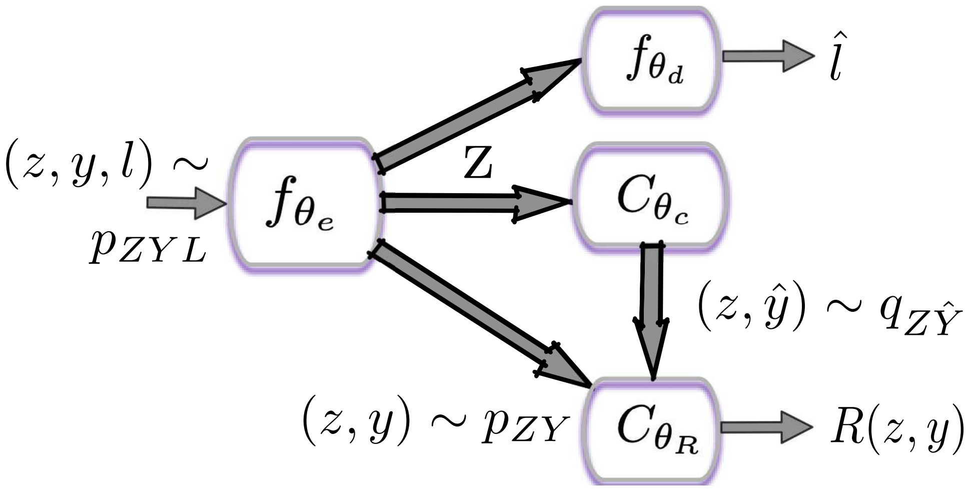
For disentanglement, we follow common practice (Lample et al., 2018) and implement a two layers perceptron (Rosenblatt, 1958). We use LeakyRelu (Xu et al., 2015) as activation functions and set the dropout (Srivastava et al., 2014) rate to 0.1.

D Additional Results on Sentiment

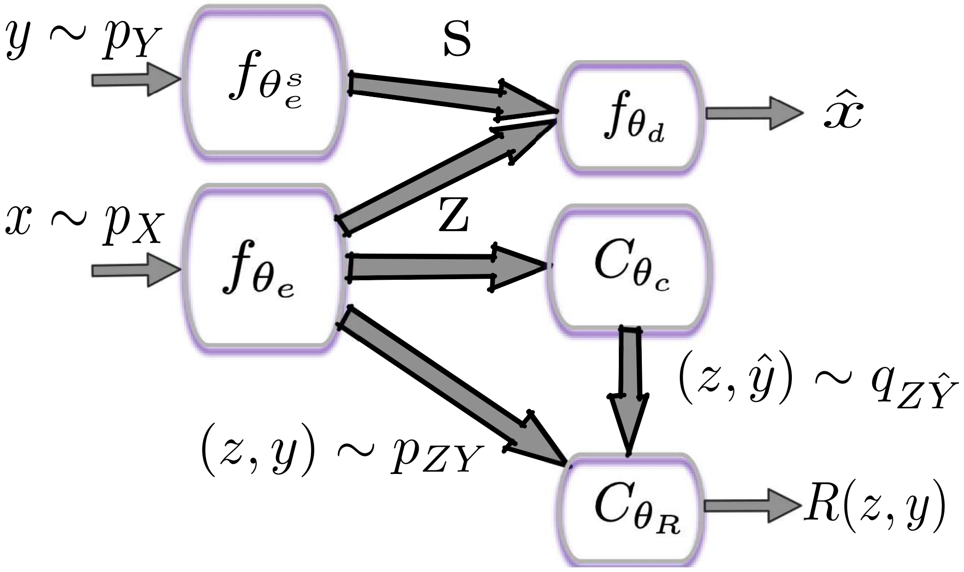
D.1 Binary Sentence Generation

D.1.1 Human Evaluation

In Tab. 1, we report the performances of systems when evaluated by humans on the polarity transfer task. 100 sentences are generated by each system and 3 english native speakers are asked to annotate each sentence along 3 dimensions (*i.e* fluency, sentiment and content preservation). Turkers assign

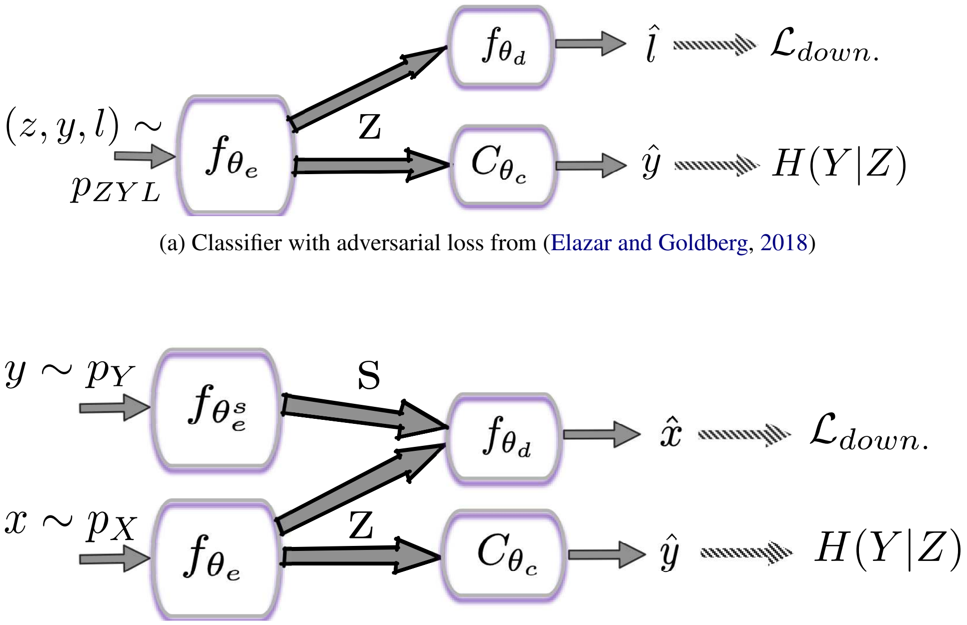


(a) Classifier with our MI surrogate



(b) StyleEmb model from (John et al., 2018) with our MI surrogate

Figure 6: Proposed methods. As described in Th. 1.



(a) Classifier with adversarial loss from (Elazar and Goldberg, 2018)

(b) StyleEmb model from (John et al., 2018)

Figure 7: Baselines methods, theses models use an adversarial loss for disentanglement. f_{θ_e} represents the input sentence encoder; $f_{\theta_e}^s$ denotes the style encoder (only used for sentence generation tasks); C_{θ_c} represents the adversarial classifier; f_{θ_d} represents the decoder that can be either a classifier (Fig. 7a or a sequence decoder (Fig. 7b). Schemes of our proposed models are given in **??**

binary labels to fluency and sentiment (following the protocol introduced in Jalalzai et al. (2020)) while content is evaluated on a likert scale from 1-5. For content preservation, both the input sentence and the generated sentence are provided to the turker. The annotator agreement is measure by the Krippendorff Alpha[2] (Krippendorff, 2018). The Krippendorff Alpha is: $\alpha = 0.54$ on the sentiment classification, $\alpha = 0.20$ for fluency and $\alpha = 0.18$ for content preservation.

D.2 Content preservation using Cosine Similarity

Fig. 8 measures the content preservation measured using cosine similarity for the sentence generation task using sentiment labels. As with the BLEU score, we observe that as the learnt representation becomes more entangled (λ increases) less content is preserved. Similarly to BLEU the model using the KL bound conserves outperforms other models in terms of content preservation for $\lambda > 5$.

D.3 Example of generated sentences

Tab. 2 gathers some sentences generated by the different sentences for different values of λ.

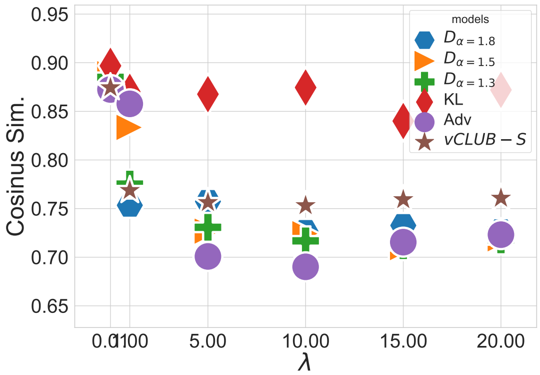


Figure 8: Content preservation measured by the cosine similarity.

Style transfert. From Tab. 2, we can observe that the impact of disentanglement on a qualitative point of view. For small values of λ the models struggle to do the style transfer (see example 2 for instance). As λ increases disentanglement becomes easier, however, the content becomes more generic which is a known problem (see (Li et al., 2015) for instance).

Example of "degeneracy" for large values of λ. For sentences generated with the baseline model a repetition phenomenon appears for greater values of λ. For certain sentences, models ignore the style

[2] Krippendorff Alpha measures of inter-rater reliability in $[0, 1]$: 0 is perfect disagreement and 1 is perfect agreement.

Model	Fluency	Content	Sentiment
Human	0.80	3.4	0.78
Adv	0.60	2.4	0.63
$vCLUB-S$	0.62	2.6	0.65
KL	0.68	2.6	0.63
$D_{\alpha=1.3}$	0.70	2.4	0.65
$D_{\alpha=1.5}$	0.68	2.9	0.70
$D_{\alpha=1.8}$	0.76	3.0	0.58

Table 1: Human annotation of generated samples. For this comparison we rely on the sentences provided in https://github.com/rpryzant/delete_retrieve_generate. Human annotations are also provided by Li et al. (2018). We have reprocessed the provided sentence using a tokenizer based on SentencePiece (Kudo, 2018; Sennrich et al., 2016). Since there is a trade-off between automatic evaluation metrics (*i.e* BLEU, Perplexity and Accuracy of Style Transfer), we set minimum thresholds on BLEU and on style transfert accuracy. The best model that met the threshold on validation is selected. We will release–along with our code–new generated sentences for comparison.

token (*i.e.*, the sentence generated with a positive sentiment is the same as the one generated with the negative sentiment). We attribute this degeneracy to the fact that the model is only trained with (x_i, y_i) sharing the same sentiment which appears to be an intrinsic limitation of the model introduced by (John et al., 2018).

Analysis of performances of vCLUB-S Similarly to what can be observed with automatic evaluation Tab. 2 shows that the system based on vCLUB-S has only two regimes: "light" disentanglement and strong disentanglement. With light disentanglement the decoder fail at transferring the polarity and for strong disentanglement few content features remain and the system tends to output generic sentences.

E Additional Results on Multi class Sentence Generation

Results on the multi-class style transfer and on are reported in Fig. 4b Similarly than in the binary case there exists a trade-off between content preservation and style transfer accuracy. We observe that the BLEU score in this task is in a similar range than the one in the gender task, which is expected because data come from the same dataset where only the labels changed.

λ	Model	Sentence
	Input	**It's freshly made, very soft and flavorful.**
0.1	Adv	it's crispy and too nice and very flavor.
	vCLUB-S	It's freshly made, and great.
	KL	it's a huge, crispy and flavorful.
	$D_{\alpha=1.3}$	it's hard, and the flavor was flavorless.
	$D_{\alpha=1.5}$	it's very dry and not very flavorful either.
	$D_{\alpha=1.8}$	it's a good place for lunch or dinner.
	Input	it's freshly made, very soft and flavorful.
1	Adv	it's not crispy and not very flavorful flavor.
	vCLUB-S	It's bad.
	KL	it's very fresh, and very flavorful and flavor.
	$D_{\alpha=1.3}$	it's not good, but the prices are good.
	$D_{\alpha=1.5}$	it's not very good, and the service was terrible.
	$D_{\alpha=1.8}$	it was a very disappointing experience and the food was awful.
	Input	it's freshly made, very soft and flavorful.
5	Adv	i hate this place.
	vCLUB-S	i hate it.
	KL	it's very fresh, flavorful and flavorful.
	$D_{\alpha=1.3}$	it's not worth the money, but it was wrong.
	$D_{\alpha=1.5}$	it's not worth the price, but not worth it.
	$D_{\alpha=1.8}$	it's hard to find, and this place is horrible.
	Input	it's freshly made, very soft and flavorful.
10	Adv	i hate this place.
	vCLUB-S	i hate it.
	KL	it's a little warm and very flavorful flavor.
	$D_{\alpha=1.3}$	it was a little overpriced and not very good.
	$D_{\alpha=1.5}$	it's a shame, and the service is horrible.
	$D_{\alpha=1.8}$	it's not worth the $ NUM.

Table 2: Sequences generated by the different models on the binary sentiment transfer task.

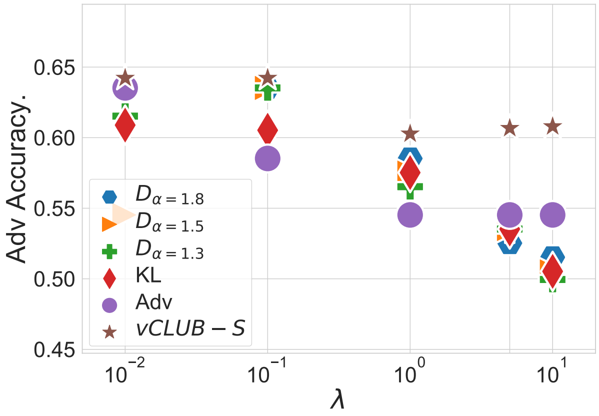


Figure 9: Disentanglement of the learnt embedding when training an off-line adversarial classifier for the sentence generation with gender data.

F Binary Sentence Generation: Application to Gender Data

F.1 Quality of the Disentanglement

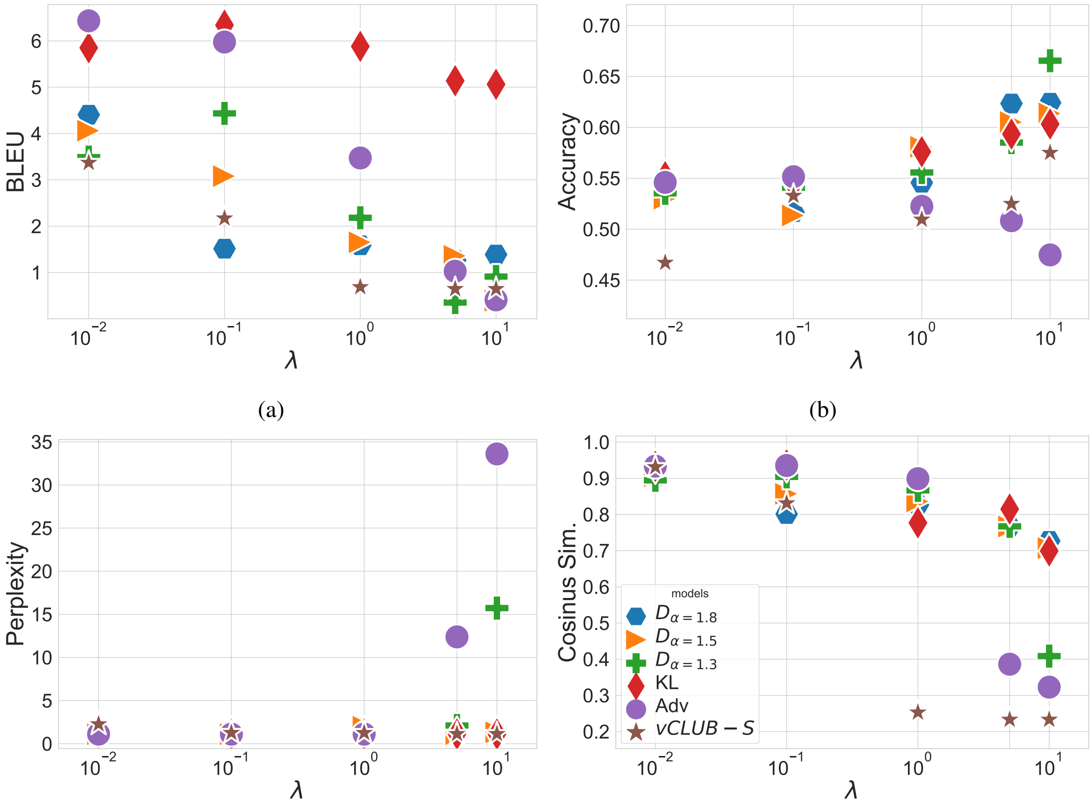
In Fig. 9, we report the adversary accuracy of the different methods for the values of λ. It is worth noting that gender labels are noisier than sentiment labels (Lample et al., 2018). We observe that the adversarial loss saturates at 55% where a model trained on MI bounds can achieve a better disentanglement. Additionally, the models trained with MI bounds allow better control of the desired degree of disentanglement.

F.2 Quality of Generated Sentences

Results on the sentence generation tasks are reported in Fig. 10 and in Fig. 11. We observe that for $\lambda > 1$ the adversarial loss degenerates as observe in the sentiment experiments.Compared to sentiment score we observe a lower score of BLEU which can be explained by the length of the review in the FYelp dataset. On the other hand, we observe a similar trade-off between style transfer accuracy and content preservation in the non degenerated case: as style transfer accuracy increases, content preservation decreases. Overall, we remark a behaviour similar to the one we observe in sentiment experiments.

G Additional Results on Multi class Sentence Generation

Results on the multi-class style transfer and on conditional sentence generation are reported in Fig. 4b and **??**. Similarly than in the binary case there exists a trade-off between content preservation and style transfer accuracy. We observe that the BLEU score in this task is in a similar range than the one in the gender task, which is expected because data come from the same dataset where only the labels changed.



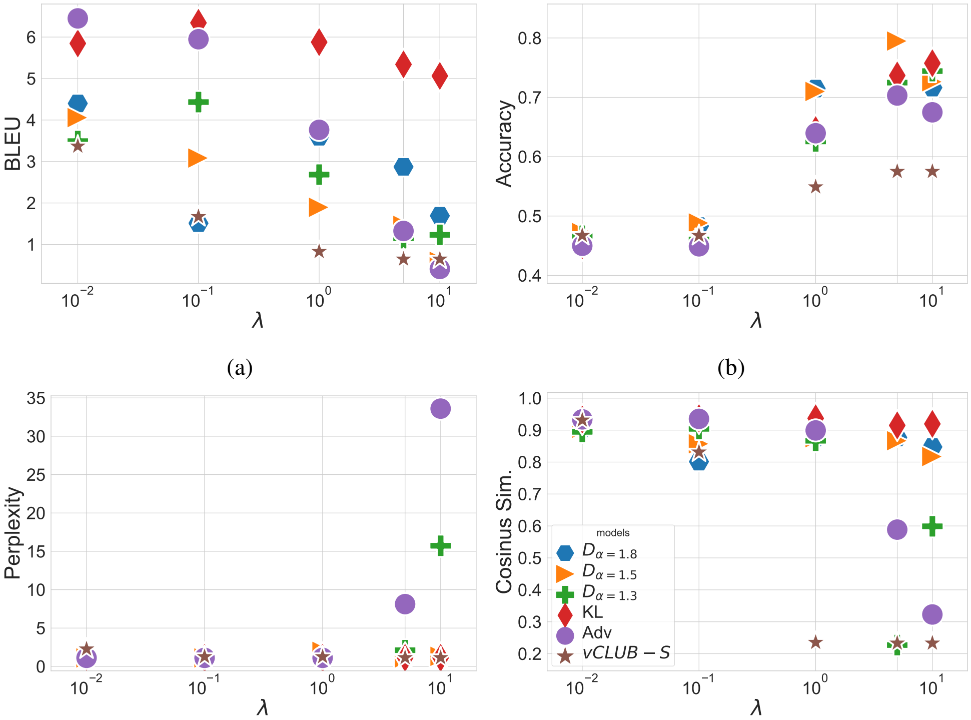
(a)

(b)

(c)

(d)

Figure 10: Numerical experiments on binary style transfer using gender labels. Results include: BLEU (Fig. 10a); cosine similarity (Fig. 10d); style transfer accuracy (Fig. 10b); sentence fluency (Fig. 10c).



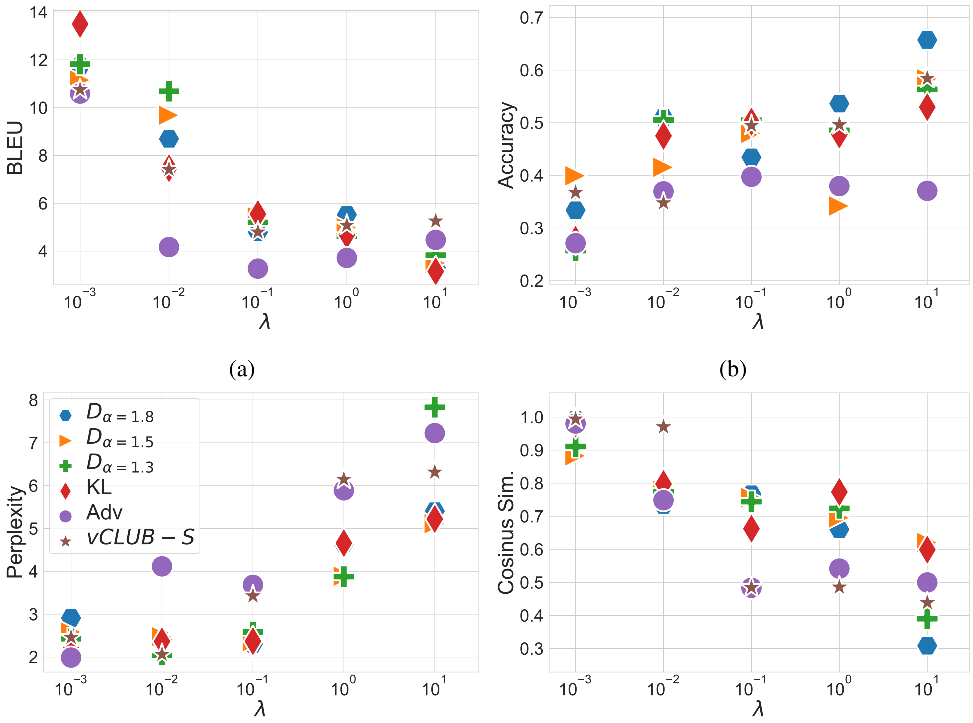
(a)

(b)

(c)

(d)

Figure 11: Numerical experiments on conditional sentence generation using gender labels. Results includes: BLEU (Fig. 11a); cosine similarity (Fig. 11d); style transfer accuracy (Fig. 11b); sentence fluency (Fig. 11c).



(a) (b)

(c) (d)

Figure 12: Numerical experiments on the multi-class conditionnal sentence generation. Results include: BLEU (Fig. 12a); cosine similarity (Fig. 12d); style transfer accuracy (Fig. 12b); sentence fluency (Fig. 12c).