读了几篇文本风格迁移的文章，简单记录：

1. Style transfer from non-parallel text by cross-alignment

认为文本中包含两种信息：内容（语义）信息和风格信息，假设不同的 corpora 具有不同的风格信息，但内容信息具有“可对齐的分布”，因此可以通过嵌入到同一个隐空间来保留与风格不相关的内容信息。

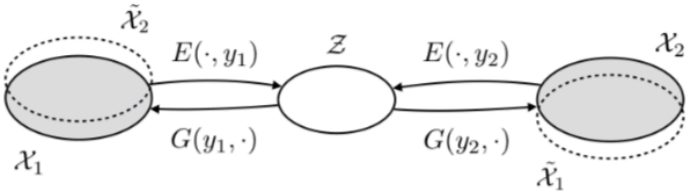

Figure 1: An overview of the proposed cross-alignment method. $\mathcal{X}_1$ and $\mathcal{X}_2$ are two sentence domains with different styles y_1 and y_2, and $\mathcal{Z}$ is the shared latent content space. Encoder E maps a sentence to its content representation, and generator G generates the sentence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

文章这样建模文本生成的过程：

1. a latent style variable $\boldsymbol{y}$ is generated from some distribution $p(\boldsymbol{y})$;
2. a latent content variable $\boldsymbol{z}$ is generated from some distribution $p(\boldsymbol{z})$;
3. a datapoint $\boldsymbol{x}$ is generated from conditional distribution $p(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{z})$.

既然假设了不同文本集的内容分布是相同的，我们用同一个 z 表示内容随机变量，对于不同的风格 y1 和 y2，以概率 p(x1|y1)和 p(x2|y2)生成两个文本集，则文本风格迁移的目标是条件概率$p(\boldsymbol{x}_1|\boldsymbol{x}_2; \boldsymbol{y}_1, \boldsymbol{y}_2)$和$p(\boldsymbol{x}_2|\boldsymbol{x}_1; \boldsymbol{y}_1, \boldsymbol{y}_2)$。

文章先给了一个结论，只要 y 不同，则 p(x1|y1)和 p(x2|y2)就不能是相同的分布，在此前提下这一目标才能 work。直觉上，不同的风格生成的文本应该有足够的差异，否则 transfer 就没有意义了。举例来说，如果内容变量 z 产生自 01 高斯分布，若风格变量$\boldsymbol{y} = (\boldsymbol{A}, \boldsymbol{b})$是仿射变换的参数 $\boldsymbol{x} = \boldsymbol{A}\boldsymbol{z} + \boldsymbol{b} + \boldsymbol{\epsilon}$，则任何正交矩阵 A，$\boldsymbol{y} = (\boldsymbol{A}, \boldsymbol{0})$都产生相同的 x 的分布---01 高斯分布。而有趣的是如果 z 采样自混合高斯分布，则对于不同的风格变量 y，产生的数据 x 是有区别的。这启示我们 z 应该有足够的复杂性，保留文本中足够多的信息，而 y 的影响应该相对小一些。

由于 x1、x2 在 z 下条件独立，则目标分布可以写为：

$$\begin{aligned} p(\boldsymbol{x}_1|\boldsymbol{x}_2; \boldsymbol{y}_1, \boldsymbol{y}_2) &= \int_{\boldsymbol{z}} p(\boldsymbol{x}_1, \boldsymbol{z}|\boldsymbol{x}_2; \boldsymbol{y}_1, \boldsymbol{y}_2) d\boldsymbol{z} \\ &= \int_{\boldsymbol{z}} p(\boldsymbol{z}|\boldsymbol{x}_2, \boldsymbol{y}_2) \cdot p(\boldsymbol{x}_1|\boldsymbol{y}_1, \boldsymbol{z}) d\boldsymbol{z} \\ &= \mathbb{E}_{\boldsymbol{z} \sim p(\boldsymbol{z}|\boldsymbol{x}_2, \boldsymbol{y}_2)}[p(\boldsymbol{x}_1|\boldsymbol{y}_1, \boldsymbol{z})] \end{aligned}$$

观察形式类似于 AE，用两个网络分别建模 p(z|x2,y2)和 p(x1|y1,z)作为 encoder 和 decoder。由于我们的数据是 unparalleled，我们不能直接用上式作为 loss----p(x1|y1,z)中某个 z 和某个 x1 并不是对应的。

首先需要隐层 z 能学到文本的信息，定义重建误差：

$$\mathcal{L}_{\text{rec}}(\boldsymbol{\theta}_E, \boldsymbol{\theta}_G) = \mathbb{E}_{\boldsymbol{x}_1 \sim \boldsymbol{X}_1}[-\log p_G(\boldsymbol{x}_1 | \boldsymbol{y}_1, E(\boldsymbol{x}_1, \boldsymbol{y}_1))] + \mathbb{E}_{\boldsymbol{x}_2 \sim \boldsymbol{X}_2}[-\log p_G(\boldsymbol{x}_2 | \boldsymbol{y}_2, E(\boldsymbol{x}_2, \boldsymbol{y}_2))]$$

同时 z 对不同数据集应该对齐，文章讨论了基于 VAE 的方法，基于前面的结论，我们需要 z 保留足够多的信息，而不是让 decoder 网络承担从简单分布生成复杂文本的能力，因此像原始的 VAE 去趋近一个高斯分布的先验是不合理的。文章讨论了两种变体，第一种就是直接在隐层对齐：

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} \mathcal{L}_{\text{rec}}(\boldsymbol{\theta}_E, \boldsymbol{\theta}_G)$$

$$\text{s.t.} \quad E(\boldsymbol{x}_1, \boldsymbol{y}_1) \overset{\text{d}}{=} E(\boldsymbol{x}_2, \boldsymbol{y}_2) \qquad \boldsymbol{x}_1 \sim \boldsymbol{X}_1, \boldsymbol{x}_2 \sim \boldsymbol{X}_2$$

文章没有严格解该最优化问题，而是直接用一个判别器来对抗的学习：

$$\mathcal{L}_{\text{adv}}(\boldsymbol{\theta}_E, \boldsymbol{\theta}_D) = \mathbb{E}_{\boldsymbol{x}_1 \sim \boldsymbol{X}_1}[-\log D(E(\boldsymbol{x}_1, \boldsymbol{y}_1))] + \mathbb{E}_{\boldsymbol{x}_2 \sim \boldsymbol{X}_2}[-\log(1 - D(E(\boldsymbol{x}_2, \boldsymbol{y}_2)))]$$

$$\min_{E,G} \max_{D} \mathcal{L}_{\text{rec}} - \lambda \mathcal{L}_{\text{adv}}$$

第二种方法直接使生成的文本对齐：$p(\boldsymbol{x}_2|\boldsymbol{y}_2) = \int_{\boldsymbol{x}_1} p(\boldsymbol{x}_2|\boldsymbol{x}_1; \boldsymbol{y}_1, \boldsymbol{y}_2) p(\boldsymbol{x}_1|\boldsymbol{y}_1) d\boldsymbol{x}_1$

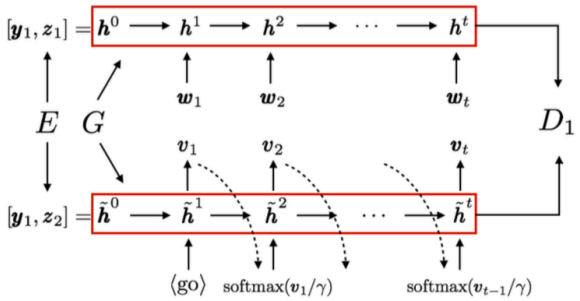

Figure 2: Cross-aligning between $\boldsymbol{x}_1$ and transferred $\boldsymbol{x}_2$. For $\boldsymbol{x}_1$, G is teacher-forced by its words $\boldsymbol{w}_1 \boldsymbol{w}_2 \cdots \boldsymbol{w}_t$. For transfered $\boldsymbol{x}_2$, G is self-fed by previous output logits. The sequence of hidden states $\boldsymbol{h}^0, \cdots, \boldsymbol{h}^t$ and $\tilde{\boldsymbol{h}}^0, \cdots, \tilde{\boldsymbol{h}}^t$ are passed to discriminator D_1 to be aligned. Note that our first variant aligned auto-encoder is a special case of this, where only $\boldsymbol{h}^0$ and $\tilde{\boldsymbol{h}}^0$, i.e. $\boldsymbol{z}_1$ and $\boldsymbol{z}_2$, are aligned.

实现上，E 和 G 两部分网络都使用 GRU，风格信息作为 encoder 网络的初始 hidden state。第二种方法中，由于 D 接在文本空间上，对于离散空间采样不可导问题，使用 gumbel softmax 方法。

2. Style transfer in text：exploration and evalution

和上一篇一样，这篇的动机仍是去学习只包含内容信息的表示，然后从中恢复出不同风格的文本。

基于 AE，为了将内容信息从文本中提取出来，本文提出了两种方法：

第一种类似于上一篇在隐层去做对抗，不同于上一篇使用的二分类器作为判别器，这一篇是训练了一个多分类器（从而可以将对多种风格的文本压缩到一个隐空间），训练分类器时将类别的负似然概率作为目标，训练 encoder 则以使分类的熵增加为目标。

$$L_{adv1}(\Theta_c) = -\sum_{i=1}^{M} \log\ p(l_i|Encoder(\mathbf{x}_i;\Theta_e);\Theta_c)$$

$$L_{adv2}(\Theta_e) = -\sum_{i=1}^{M}\sum_{j=1}^{N} H(p(j|Encoder(\mathbf{x}_i;\Theta_e);\Theta_c))$$

文章使用了多个生成器（decoder）来恢复不同风格的文本，loss 就是重建误差：

$$L_{gen1}(\Theta_e,\Theta_d) = \sum_{i=1}^{L} L^i_{seq2seq}\left(\Theta_e,\Theta^i_d\right)$$

第二种方法仅在生成的部分有所差别，选择将 style 信息作为 decoder 的输入，值得一提的是 style 信息并不是用一个 indicator 或者随机初始化一些表示不同 style 的向量，而是预训练了 style embedding。

3. Toward controlled generation of text

同样基于 VAE，这篇和前两篇的区别在于直接的建模了风格变量 c 的分布，试图学到相互独立的内容变量 z 和风格变量 c。整个模型框架如下图所示

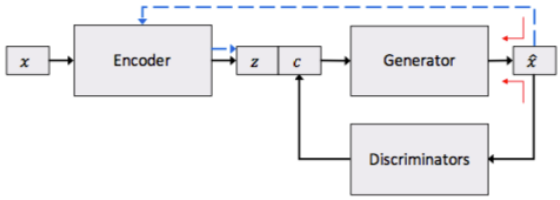

Figure 1. The generative model, where $\boldsymbol{z}$ is unstructured latent code and $\boldsymbol{c}$ is structured code targeting sentence attributes to control. Blue dashed arrows denote the proposed independency constraint (section 3.2 for details), and red arrows denote gradient propagation enabled by the differentiable approximation.

其中从 x→z→x'是一个 VAE 模型，z 由 encoder 网络产生，以高斯分布为先验；c 则由另外的一个网络（discriminator）产生，而之所以不用另外一个 encoder、以 VAE 的 loss 建模 c 是因为往往我们会选择离散变量作为 c，而建模 p(c|x)实际上是要训练一个分类器，我们先关注 VAE 的部分：

$$\begin{aligned}\mathcal{L}_{\text{VAE}}(\boldsymbol{\theta}_G,\boldsymbol{\theta}_E;\boldsymbol{x}) &= \text{KL}(q_E(\boldsymbol{z}|\boldsymbol{x})\|p(\boldsymbol{z})) \\ &\quad - \mathbb{E}_{q_E(\boldsymbol{z}|\boldsymbol{x})q_D(\boldsymbol{c}|\boldsymbol{x})}\left[\log p_G(\boldsymbol{x}|\boldsymbol{z},\boldsymbol{c})\right],\end{aligned} \tag{4}$$

为了保证 G 能够利用 c 中的信息成功 transfer，我们可以选择在文本 x 所在空间或者特征 c 所在空间做约束，本文认为在特征空间会提供更好的度量，所以训练 G 使得生成的 x'可以被 D 正确的分类：

$$\mathcal{L}_{\text{Attr},c}(\boldsymbol{\theta}_G) = -\mathbb{E}_{p(\boldsymbol{z})p(\boldsymbol{c})}\left[\log q_D(\boldsymbol{c}|\widetilde{G}_\tau(\boldsymbol{z},\boldsymbol{c}))\right]. \tag{6}$$

此外，为了保证 c 中只包含和指定属性（比如情感）有关的信息，我们需要 z 中保留其他所有的信息，为达到这个目的，将 encoder 作为判别器，训练 G 使得重新 encode 的隐变量和原来的 z 一致：

$$\mathcal{L}_{\text{Attr},z}(\boldsymbol{\theta}_G) = -\mathbb{E}_{p(\boldsymbol{z})p(\boldsymbol{c})}\left[\log q_E(\boldsymbol{z}|\widetilde{G}_\tau(\boldsymbol{z},\boldsymbol{c}))\right]. \tag{7}$$

最后 G 的 loss 包含如下几项：

$$\min_{\theta_G} \mathcal{L}_G = \mathcal{L}_{\text{VAE}} + \lambda_c \mathcal{L}_{\text{Attr},c} + \lambda_z \mathcal{L}_{\text{Attr},z}, \qquad (8)$$

下面看 D 部分：

$$\mathcal{L}_u(\boldsymbol{\theta}_D) = -\mathbb{E}_{p_G(\hat{\boldsymbol{x}}|\boldsymbol{z},\boldsymbol{c})p(\boldsymbol{z})p(\boldsymbol{c})}\left[\log q_D(\boldsymbol{c}|\hat{\boldsymbol{x}}) + \beta \mathcal{H}(q_D(\boldsymbol{c}'|\hat{\boldsymbol{x}}))\right] \qquad (10)$$

可以看到该目标和(6)一致，也就说虽然叫做判别器，实际上是不存在对抗的，这意味着 D 作为分类器，没有负例。因此文章使用了半监督的方式来训练，标定了一些正负样例，以下式为目标：

$$\mathcal{L}_s(\boldsymbol{\theta}_D) = -\mathbb{E}_{\mathcal{X}_L}\left[\log q_D(\boldsymbol{c}_L|\boldsymbol{x}_L)\right].$$

$$\min_{\theta_D} \mathcal{L}_D = \mathcal{L}_s + \lambda_u \mathcal{L}_u, \qquad (11)$$

整个算法如下：

Algorithm 1 Controlled Generation of Text

Input: A large corpus of unlabeled sentences $\mathcal{X} = \{\boldsymbol{x}\}$
A few sentence attribute labels $\mathcal{X}_L = \{(\boldsymbol{x}_L, \boldsymbol{c}_L)\}$
Parameters: $\lambda_c, \lambda_z, \lambda_u, \beta$ – balancing parameters

1: Initialize the base VAE by minimizing Eq.(4) on $\mathcal{X}$ with $\boldsymbol{c}$ sampled from prior $p(\boldsymbol{c})$
2: **repeat**
3: Train the discriminator D by Eq.(11)
4: Train the generator G and the encoder E by Eq.(8) and minimizing Eq.(4), respectively.
5: **until** convergence

Output: Sentence generator G conditioned on disentangled representation $(\boldsymbol{z}, \boldsymbol{c})$

4. Cycled reinforcement learning approach

比较 nlp 的一篇文章，思路是显示的学习哪些词是情感词，然后对去除情感词的句子进行重建，模型启动需要预训练。

模型分为两部分，中立模块对句子中的每个词输出是否是情感词的二分类结果，预训练用 self-attention 模型对句子进行消极/积极分类，这个过程中学到的了每个词的 attention 权重，用该值作为是否是情感词的分类依据。

情感模块是一个 encoder-decoder 模型，输入为去除情感词的句子，输出为原始句子，所以可以利用中立模型有监督的去训练。

反过来，情感模块对同一个输入产生两个情感相反的句子，用该输出定义了指导中立模块学习的 reward（基于 policy gradient），包含两个部分，分别考察情感转换度和内容保留度。

总体来说，该模型并不是像 cycleGAN 一样在积极和消极之间来回转换，而是：学习情感中立—重建情感句子这两步相互指导，涉及到一些比较 nlp 的方法，不太具有借鉴价值。另外 reward 的第二部分使用 bleu 值来评价内容保持程度的，而最终的结果评价也是用 ble u 值，感觉有些问题。

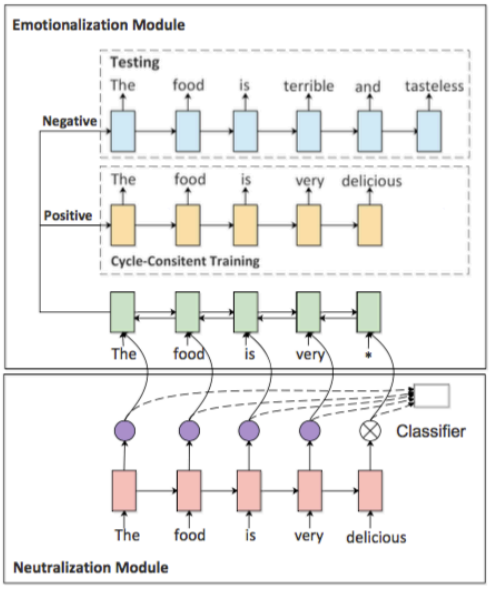

Figure 1: An illustration of the two modules. Lower: The neutralization module removes emotional words and extracts non-emotional semantic information. Upper: The emotionalization module adds sentiment to the semantic content. The proposed self-attention based sentiment classifier is used to guide the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

5．评估指标

评估文本风格迁移的质量一般要考虑两个方面：transfer 的能力和语义信息的保留。

风格迁移任务：

考察 transfer 的效果，在 yelp 餐厅评价系统中，用户会对餐厅进行评分同时发表评价，使用这些评价作为训练语料，评分来区分句子是 positive 还是 negative。评估时，通过预训练的分类器，最后看有多大比例的 transfer 后的句子是否能够骗过分类器。[1]在此任务上增加了人为评分，分别考察 transfer 后句子的流畅性、风格是否成功的 transfer、以及语义内容是否得以保留。[2]通过计算 transfer 前后句子 embedding 的 cosine 距离来评估句子是否保留了语义信息：

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

单词替换解密任务：

各单词一对一的进行替换，从而从明码文本得到加密的文本，解密任务是不知道替换键的情况下，从加密文本解密回原始文本。为了得到 unparalleled 的数据，

分别从 yelp 数据集中取两堆句子，对其中一堆进行加密，作为训练集，测试时，由于有 parallel 的文本，所以用 Bleu 值来评估。

单词顺序恢复：

类似于上一个任务，通过某种固定的方式对句子中的词序进行打乱作为新的风格，通过模型来学习乱序和正确顺序之间的 transfer。