

Acl2021

文本风格转换的目的是在保留内容的同时改变句子的风格(如情感)。一种常见的方法是将给定的句子映射到没有样式的内容表示，然后将内容表示提供给具有目标样式的解码器。以前的过滤样式的方法在标记级别完全删除带有样式的标记，这会导致内容信息的丢失。在本文中，我们建议通过反向注意隐式删除每个标记的样式信息来增强内容保存，从而保留内容。此外，我们在构建目标样式表示时融合了内容信息，使其相对于内容具有动态性。我们的方法不仅创建了与样式无关的内容表示，而且还在转移样式中创建了与内容相关的样式表示。实验结果表明，我们的方法在内容保存方面大大优于目前最先进的基线。此外，它在风格转换的准确性和流畅性方面也具有竞争力。

风格转换是计算机视觉和自然语言处理中的一项热门任务。它的目的是将具有某种风格的输入(如情感、形式)转换为不同的风格，同时保留原始内容。

本文提出了一种新颖的文本样式转换方法。一个关键的想法是利用标记通常同时拥有样式和内容信息这一事实。例如，单词“delicious”是一个带有强烈风格信息的标记，但它也暗示主题是食物。这些词在表现风格(如积极情绪)和暗示主题/内容(如食物)方面发挥着关键作用。完全删除这些标记会导致内容信息的丢失。

关键点1：

在风格分类任务中，注意得分可以解释为标记具有风格属性的程度。如果我们能够确定哪些标记揭示了风格属性，以及在何种程度上揭示了风格属性，那么就有可能获得否定并接近标记中内容属性的数量。在本文中，我们称之为反向注意。我们利用这样的分数来抑制标记的风格属性，充分捕捉内容属性。

关键点2：

本文在创建目标样式表示时，通过融合内容信息进一步增强内容保存。尽管在创建内容表示方面做了大量的努力，但之前的工作忽略了构建依赖于内容的样式表示。通常的方法是将目标样式投射到嵌入空间中，并将嵌入的样式作为输入共享给解码器。然而，我们的工作通过利用条件层归一化(CLN)来阐明构建内容相关的风格。我们的这个模块接受内容表示，并通过塑造内容变量以适应目标样式的分布来创建依赖于内容的样式表示。这样，即使目标样式相同，我们的样式表示也会根据输入序列的内容而变化。

创新点：

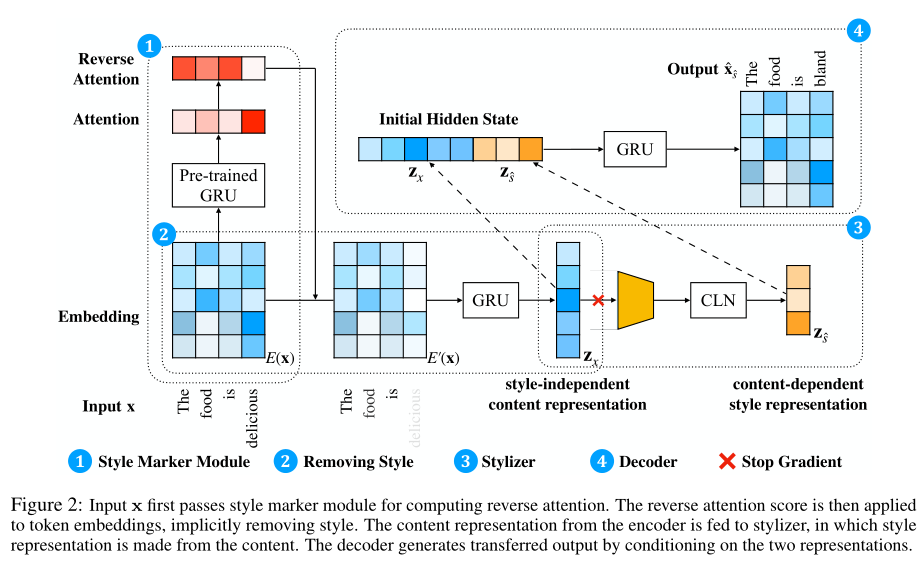
1. 我们引入反向注意，作为在构建输入的内容表示时，在保留内容信息的同时抑制样式信息的一种方法。
2. 除了构建与样式无关的内容表示，我们的方法利用条件层规范化来构建与内容相关的样式表示。
3. 模型在内容保存方面获得了最佳性能，在Yelp数据集上BLEU有4分以上，并显示在其他指标的竞争力。

形式化定义：

是一个训练语料库，每个语料库是一个句子，每个语料库的是它的风格标签。我们的实验是在一个情绪分析任务中进行的，其中有两个风格标签，即“积极”和“消极”。任务是学习具有参数的模型，该模型接受输入句子和目标样式输入，并输出一个新的句子，该句子采用目标样式并保留的内容信息。

由于是无监督学习，因此基于预训练的风格分类器s=C(x)，输入句子输出风格。我们的目标是将它错误的分类到新的分类中。

我们的模型的架构如图所示，它有时被称为生成器网络。它包括一个编码器，一个风格化和一个解码器。编码器将输入序列映射为与样式无关的表示形式。特别的，编码器有一个样式标记模块，它计算输入标记的注意分数，并将它们“反向”以估计内容信息。将反向的注意分数应用于token嵌入E(x)，结果E’(x)被反馈给双向GRU以生成。这个样式器将目标样式和内容表示作为输入，并生成一个与内容相关的样式表示。最后，解码器接受内容表示法和样式表示法作为输入，并生成一个新的序列。

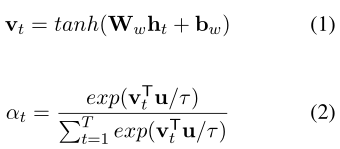


1. 风格标记模块

输入：句子x=[x1,x2,…,xT] 风格s

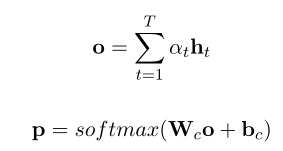
经过预训练的模型，产生风格的概率。

使用一层的双向GRU，再经过attn，具体的：



其中，ht是从时间t时刻经过双向GRU生成的表示。U是参数，是softmax的蒸馏温度。

注1：在对样式标记模块进行预训练时，我们通过取标记表示的加权和(权重为注意分数)来构造一个句子表示，并将上下文向量提供给一个完全连接的层。（从t到t+1）

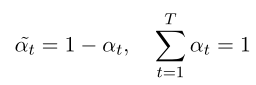
**

注2：交叉熵损失用于学习风格标记模块的参数。风格标记中的注意分数表明哪些标记对风格分类是重要的，以及重要到什么程度。这些分数将在下一节中被“反转”以显示内容信息。一旦样式标记模块经过训练，样式标记模块的全连接层就不再需要了。因此它在最后一层被移除。

综上，一个预训练的风格器就训练完成了。

1. 反转注意力

利用预先训练的风格标记模块中的注意得分，我们建议隐式地删除每个标记中的风格信息。我们通过否定每个标记中风格信息的程度来估计内容信息的程度，即反向注意。



我们将反向注意力分数乘token的嵌入。



由此你可以看到，我们隐式的删掉了风格信息，抑制了正则化的标记嵌入方面。这些表示最终流到双向GRU中，生成内容表示（即最后一个隐藏状态）。



现在，我们将句子映射为与风格无关的内容表示了。

1. 样式器

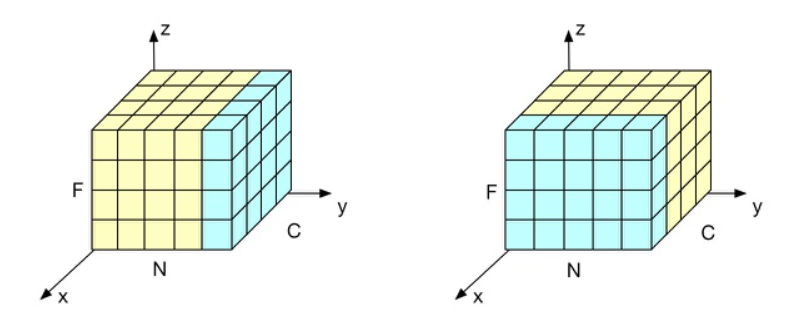
样式器的目标是创建一个与内容相关的样式表示。我们通过将编码器的内容表示形式zx应用条件层规范化来实现这一点，将其作为该模块的输入。

层规范化要求增益和偏置参数的数量与输入表示的大小相匹配。因此，主要是为了缩小大小，我们对content变量进行仿射变换。

层归一化（LN）vs BN：

LN：对样本N的通道和特征进行归一化，即有一个样本，目标是不同特征的归一化。

BN：对通道C中的样本和特征进行归一化，即有一批数据，目标是不同样本之间的归一化。



在BN和LN都能使用的场景中，BN的效果一般优于LN，原因是基于不同数据，同一特征得到的归一化特征更不容易损失信息。

但是有些场景是不能使用BN的，例如batchsize较小或者在RNN中，这时候可以选择使用LN，LN得到的模型更稳定且起到正则化的作用。RNN能应用到小批量和RNN中是因为LN的归一化统计量的计算是和batchsize没有关系的。

建议：LSTM中使用LN，MLP和CNN用BN。

公式化描述：





归一化方法通常用于改变共同尺度下的特征值，但可以保留其隐式特征值。因此，我们认为规范化的内容特征值保留了内容变量的内容信息。通过条件层归一化模块，用风格特定的增益和偏置参数对内容潜向量进行缩放和移位，落入目标风格分布。（即从已知风格到新风格，相当于整体的迁移）因此，与以前的文本样式转换尝试不同，样式表示是动态的，是依赖于内容的嵌入。

注：为了阻止与样式流进zx相关的反向传播信号，我们在把它输入样式器之前在zx上应用停止梯度。

1. 解码器

解码器根据内容相关的风格表示和内容表示生成目标风格的句子。我们使用一个单层的GRU来构造解码器。



我们的生成器的输出将进一步传递给不同的损失函数。然而，采样过程或贪婪解码不允许梯度流，因为这些方法是不可微的。因此，我们使用软采样来保持梯度流。具体来说，当输出需要梯度流时，我们取每个时间步的概率分布与嵌入层的权值的乘积，将输出投影到词嵌入空间中。我们通过实验发现，软采样比gumbel-softmax更适合我们的环境。

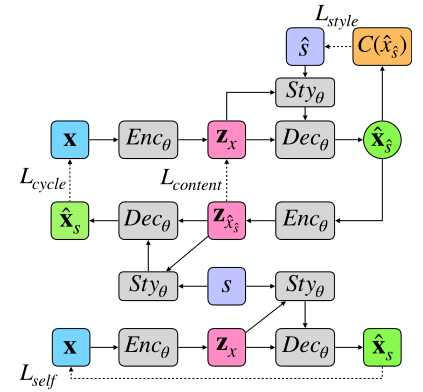
1. 预训练风格分类器

由于缺乏平行语料库，我们无法用最大似然估计训练生成器网络的风格传递能力。因此，本文采用预训练的分类器C(x)来训练我们的迁移风格。我们的分类器网络与风格标记模块具有相同的结构，但它是由一组**不同的初始模型**参数得到的独立模型。我们使用交叉熵损失进行训练：



当它被训练好后，就冻结参数。

损失函数图：



1. 自重建损失



如果重建自己的风格，应该是越相同越好。

1. 循环重建损失



转移到新风格，再转移回来，越相同越好。

1. 内容损失



还是上面的过程，内容要保持一致（不看风格）。

1. 风格迁移损失



判断迁移后的句子是不是符合风格，由风格判定器C随机生成，看是不是正确的分类。使用软采样。

由此，损失函数：



实验：数据集Yelp和IMDB

细节：

1. 嵌入128维，encoder的隐藏层维数500.
2. 条件层的正则化是200,
3. 解码器的隐藏表示层是700.
4. Adam
5. lr=0.0005
6. 前两个loss的为0.5，后两个为1

指标：

风格迁移准确度：衡量生成的句子是否揭示目标风格属性。我们之前提到过一个样式分类器:C(x)，它在loss函数中使用。为了评估传递准确性，我们训练了另一种风格分类器Ceval(x)。它具有与之前相同的架构，并使用相同的数据进行训练，除了初始模型参数的不同集合。我们使用这种结构是因为它比常用的cnn分类器性能更优越。我们的评价分类器在Yelp上的准确率达到97.8%，在IMDB上的准确率达到98.9%，高于基于cnn的准确率。

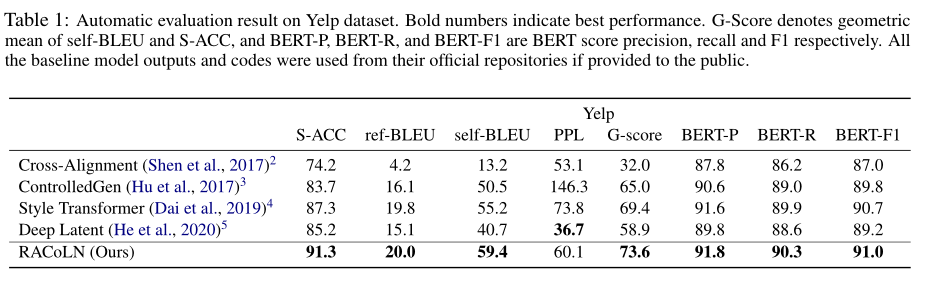
内容一致性：一个传递良好的句子必须保持其内容。在本文中，内容保存用两个BLEU评分来评估，一个是生成句和输入句之间的评分(self-BLEU)，另一个是人工生成句(ref-BLEU)。通过这个指标，我们可以评估一个句子如何在整个推理过程中维护其内容。

流畅性：自然语言生成任务的目的是输出一个句子，不仅要针对特定的任务，而且要流畅。本研究通过测量生成句子的困惑度(PPL)来衡量流畅性。接下来，我们使用在两个训练数据集上训练的5-gram KenLM。PPL得分越低，说明转移句的流利程度越高。

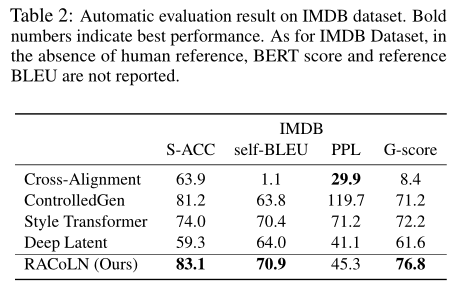
BERT评分： BERT评分计算两个句子的上下文相似度。以前的方法，如BLEU评分，计算ngram匹配评分，而BERT评分评估从预处理BERT中获得的token的上下文嵌入。该评价指标已被证明与人类判断相关，因此我们的论文包括模型生成的输出和人类参考句之间的BERT评分。我们报告精确度、召回率和F1分数。

结果：

Yelp：

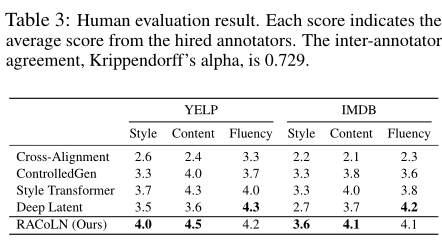


IMDB

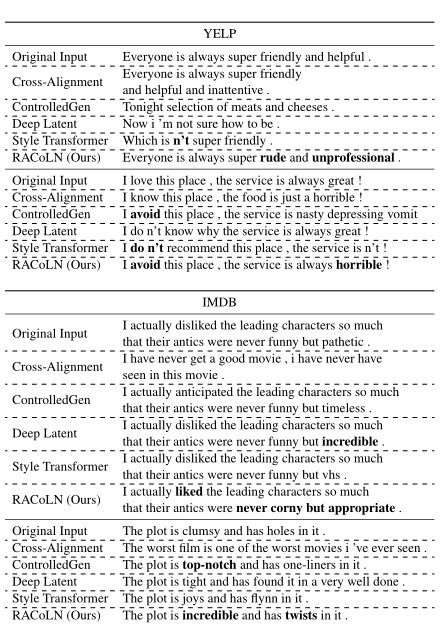


人类评估：

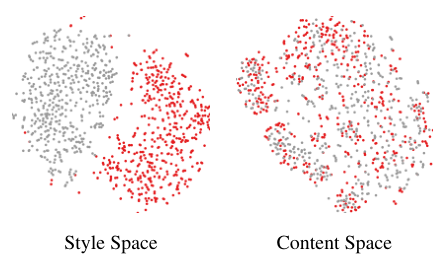
除了自动评估外，我们还使用人工评估来验证生成的输出。对于每个模型，我们从两个数据集中随机抽取150个输出，每个模型总共有300个输出。给出目标风格和原始句子，要求评注者在内容保存、风格转换准确性和流畅性方面对模型生成的句子进行评分，评分范围从1 (Very Bad)到5 (Very Good)。我们在表中报告了4个注解者的平均分数。



个例分析（加粗表示无语法错误）：



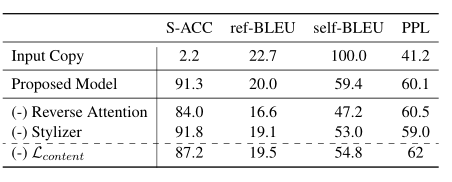
t-sne可视化：



我们在图中使用t-SNE将Yelp的测试数据集投影到内容和样式空间中。可以清楚地观察到内容表示(zx)分布在内容空间中，这表明表示是独立于样式的。在内容表示通过风格器模块之后，在样式空间中不同样式表示之间有一个明显的区别。这与图右侧所示的与样式无关的内容表示的相应分布形成了鲜明的对比。

该图清楚地描述了风格器模块中特定于样式的参数如何塑造内容表示以符合目标样式分布。该图说明了我们的模型如何成功地在编码器中删除样式，并在stylizer模块中构造与内容相关的样式。

消融实验：



我们观察到在没有反向注意模块的情况下，所有方面都有显著下降。

在另一种情况下，我们删除风格器模块，并像之前的论文一样使用样式嵌入，模型失去了保留内容的能力，在self-BLEU上下降了大约6分。

我们发现，这两个核心要素在文本风格的成功传递中是相互依存的。

最后，对于损失函数来说，加入l\_content可以显著增加内容的保存。

Gumbel-softmax：重采样技巧

如果涉及到类似选择一个行为的时候，为了让它符合概率值（而不是选最大的，argmax不可微），我们需要将随机选取的过程移出梯度图中，具体的：



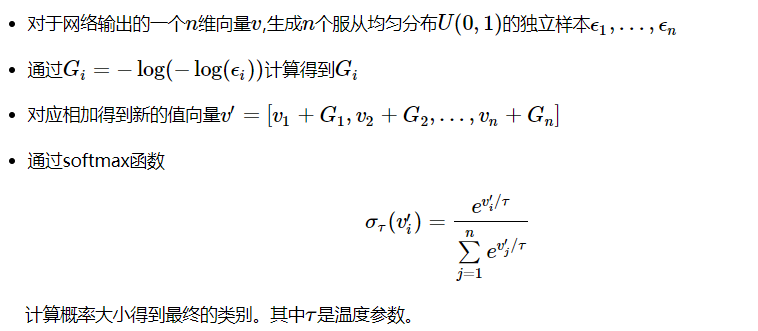
此时，不更新，仅认为是一个没有权重的输入。是一个正态分布。

例子：原始的公式：



Gumbel-CDF：

过程：



附：为什么Gumbel可以模拟出softmax这样的行为？

