基于图片关键词的古诗生成

杨雨翔 杨昆达 陈彦硕

一. 摘要

随着近年来自然语言处理领域的发展，用神经网络模型自动生成文本的方法已经越来越多地得到了人们的关注。对于各种文体，例如小说、学术论文、十四行诗甚至是中文古诗等都出现了较为成熟的生成模型。但是回顾之前的工作，大多是基于关键词或一段文本生成，信息较为单一和局限，很难完全表达出生成文本的含义，因此也就难以得到完全理想的生成结果。较为全面地生成需要视觉、文字、语音等信息的综合考量。例如对于中文古诗，诗人往往是在某些场景下触景生情而写出的，其意境更适合用图片来表现。因此本文利用两个模型的结合实现了由图片生成诗的应用，并构造了一个小规模的数据集以完成训练。同时提出了一种端到端的Encoder-Decoder模型，更好地利用图片信息进行文本生成。关于第一种方法已经有一个初版的demo，而后一种模型仍需要进一步实现和验证。

二. 方法

2.1 数据集构建

整体模型的训练依赖于一个从图片到诗的数据集。由于没有现存的合适数据集，笔者自行构建了一个小规模数据集。考虑到在图片到诗的过程中，根据图片往往只能确定一些大概的意象，我们决定从诗句中筛选出关键字，并对关键字进行图片检索，用关键字为媒介沟通图片和诗的相关性。

数据集构建的所有过程中，最重要也就是关键字的选取。在构建过程中，由于没有良好的古诗分词工具，我们就以单字分割，提取出古诗数据集[1]中的所有字，然后借用分词软件jieba[2]进行词性判断，从所有文字中汇总出出现频率最高的 100 个名词。以这 100 个名词为蓝本，再人工筛选出 30 个易于检索出高质量图片的名词，并用相关的词组代替单字进行爬虫。此时就有了图片数据集和其对应的古诗数据集。

```
Topword = [
    '人','山','水','路','雨','草','石','光','酒','雪',
    '衣','海','木','柳','霜','霞','星','眼','田','浪',
    '船','帆','室','帝','兵','手','琴','僧','紫','花',
]
```

图 1. 选择的关键字

2.2 图片特征提取

图片特征的提取使用了ResNet34[3]的结构。通过前文中构建的图片和关键词标签数据进行训练，进行softmax处理之后得到概率向量的Top-k用于下文的文本生成。这里选用的是Top-5。

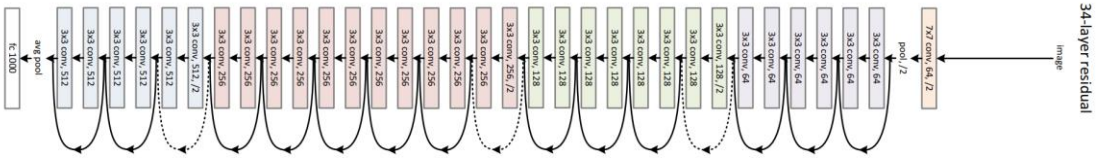

图 2. ResNet34 结构

2.3 诗句生成网络

考虑到在古诗中，往往每个字就作为一个意象单元，即一个独立的元素参与诗句意象的组织，因此选定了使用字符级别的循环神经网络[4]来进行诗句的生成。具体来说，此模型会根据

当前句子中的前面几个汉字的信息，从而预测下一个字，逐个字地进行生成，最终形成一首诗句。而生成所使用的第一个关键词就来自上文中提取到的图片信息。

下图是大致的网络结构。为了更好地利用前文信息，选用了LSTM作为循环神经网络的基本单元，输入首先经过 4 层LSTM网络之后，再经过 4 个全连接层，最后输出当前预测的字符。

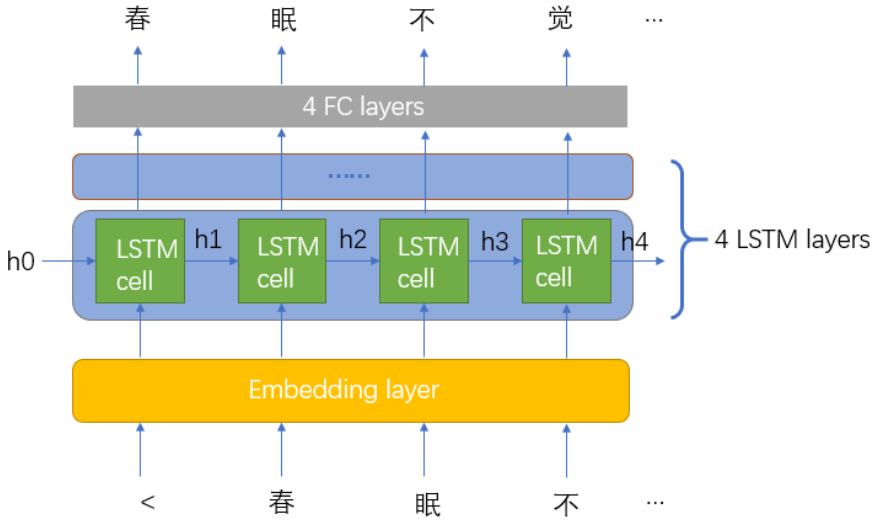

图 3. 网络结构

相应地，在训练过程中我们使用teacher forcing的方式。即在训练时，下一时刻LSTM单元的输入总为上一步的标签。最终模型在训练集上达到了 85%的top5 准确率。

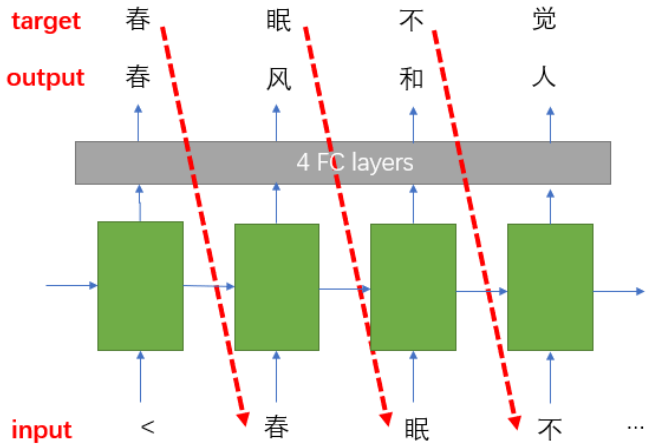

图 4. Teacher forcing训练方法示意图

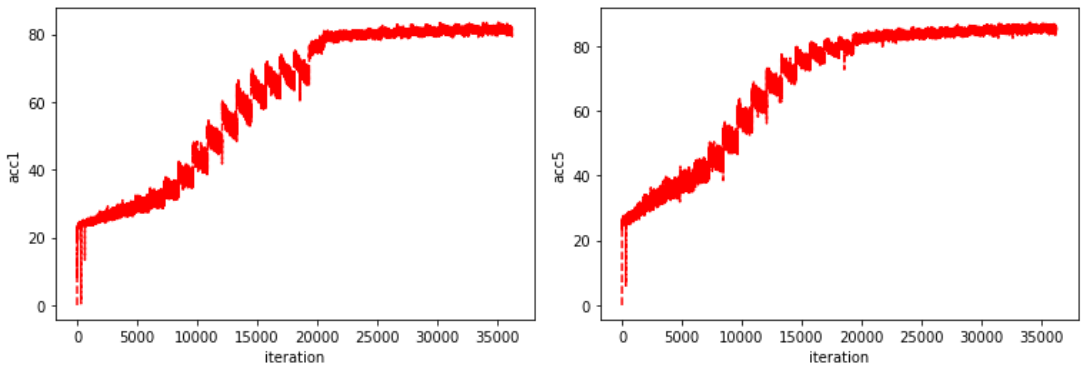

图 5. 准确率（左侧为top1 准确率，右侧为top5 准确率）

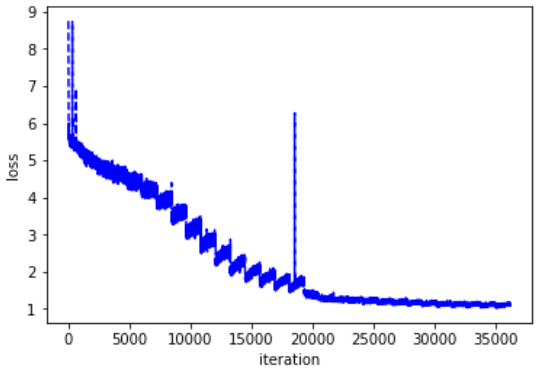

图 6. 损失函数

2.4 诗句生成器的具体实现

在诗句生成过程中，为了使生成结果更加拟真，需要对该过程引入随机性。即当我们输入两个类似的图片时，我们不期望输出相同的诗句。

这里在两个方面引入随机性：第一处是在通过图片分类器得到top-k标签时。我们利用标签生成第一句的时候使用top1 的关键词进行启发，这一步是固定的。但是在生成第二句的时候，则随机地选取其他top-k的标签。

第二处引入随机性的地方在于，当预测每一个字符的时候，我们并不一定选择概率向量的top1 作为结果，而是根据概率从高到低进行随机选词。经过尝试，最后选择top3 进行取词，可以较好地获得诗歌生成质量和随机性的平衡。

因为生成的目标是古代诗歌，因此押韵是一个重要的部分。但是仅仅依靠诗句生成模型，则较难完成押韵的工作。因此，在生成后半句的最后一个字的时候，我们额外地进行了押韵处理，根据押韵规则进行选字。

下图展示了具体的选择方法。在生成最后一个字的时候，我们不局限于top3 选词，而是取一个更大的topk，由概率从大到小依次考察每个字是否与上半句的最后一个字在平仄和韵母上构成押韵，并以此进行选择。同时为了保证质量，生成器不会过于追求押韵，当在topk中找不到符合要求的字时，依然使用top3 选词。

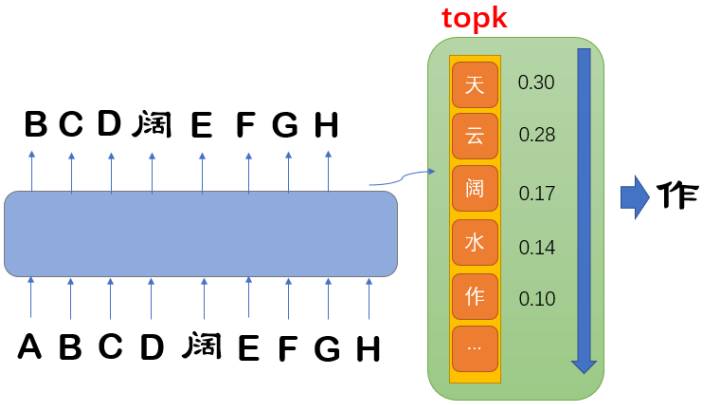

图 7. 选词方式

三. 结果

此模型的作诗效果并不好，有各种各样的问题。有时候会出现诗句顺畅但文不对图，意境有些偏离的情况；有时候又会出现诗句质量不高，读起来不顺畅的情况；仅有少数情况下会产出耐人寻味的诗句。这些问题与模型结构有很大的关系，另外与数据集不充分也有很大关系。

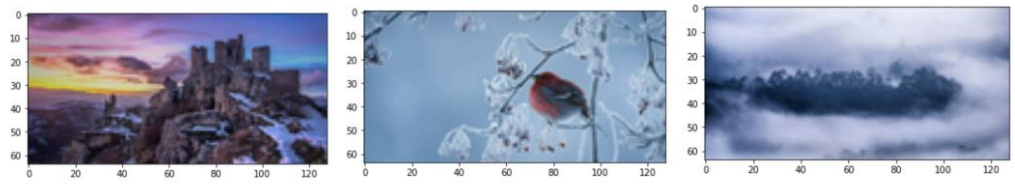

图 8. 诗句生成实例（左中两个效果较差，右边效果较好）

四. 讨论

上文实现的方法只是一个简单的流程，距离较成熟地使用仍有很长距离。主要有以下的不足：

1. 模型不是端到端的，故生成结果很难和输入产生直接的关系。前文使用图片识别的关键词来生成古诗，因此生成的结果和识别图片的种类直接相关。如果生成结果不佳，其反馈很难对前一部分的图片识别模块产生作用，即无法影响图像识别模型的参数。

2. 由于模型的分散性，导致图片的信息产生丢失。原本丰富的图像和模式信息被转换为一些离散的关键词，诗句即根据这些关键词生成，导致信息量被大大压缩。

3. 图像识别的部分使用了ResNet模型，训练时以单标签的结果进行训练。但是在使用过程中选择了Top-k的关键字，而余下的关键字是没有被考虑在损失函数中的，因此预测的结果可能有较大出入。由于图片信息得到的只有这数个关键词，因此有可能出现完全混乱的结果。

4. 训练集过于单一。由于训练图片是根据古诗关键字进行爬取，因此图片带有很强的特征性，主要集中于自然景观和古建筑、古人物等。模型在这几类图片中表现较好，但是在其他类型的图片中则较难达到好的效果。

五. 展望

为了解决当前模型的不足，我们还提出了一种端到端的模型，可以更好地利用图片的信息进行训练。模型的结构如下图所示，主要包含Encoder和Decoder两部分。

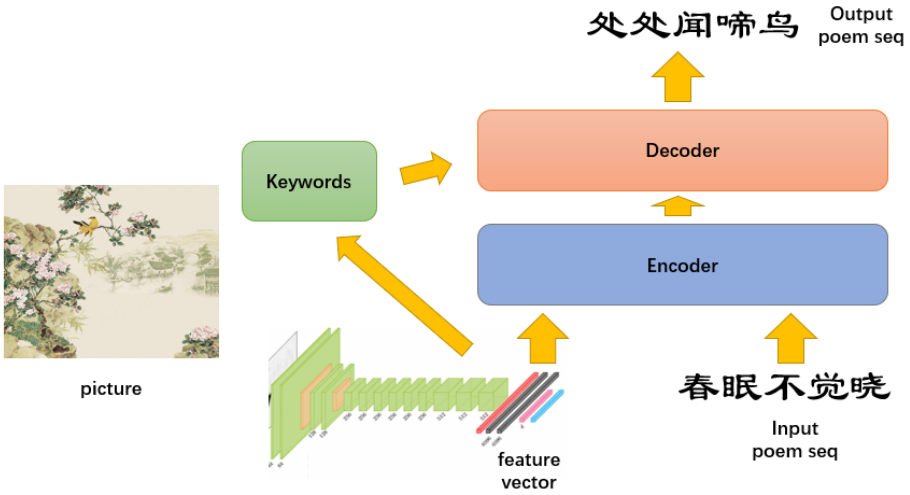

图 9. Encoder-Decoder模型

1. 图片特征提取

第一部分是对图片进行处理。相较于之前不同的是，这里是直接将网络输出的结果组织成一个向量而非利用其结果，这样可以更好地保留原有的信息。关于模型的选择，可以使用预训

练的一些CNN模型，例如ResNet，ShuffleNet等。提取的特征向量记作$V(P)$，同时利用模型输出另一组特征向量作为Decoder的输入，记作$K(P)$。

2. Encoder

Encoder使用RNN的结构，接受图片特征$V(P)$和上一句生成的结果L_{i-1}以及上一时刻的序列S_{i-1}作为输入，生成一个序列S_i。可以有以下的表示

$$S_i = f(S_{i-1}, V(P), K(P), L_{i-1})$$

3. Decoder

Dcoder同样使用RNN的结构，接受中间序列S_i和图片关键向量$K(P)$和上一句的输出L_{i-1}作为输入，并产生一个在词典上的概率向量，利用Softmax生成结果。可以有表示：

$$L_i = g(K(P), S_i, L_{i-1})$$

如此可以实现一个端到端的模型，充分利用图片的信息进行训练。由于时间原因，此想法尚未完成，暂时使用了前文提到的模型。在下一步的训练中可以利用前文模型作为训练数据生成器，快速地建立出大量的图片到诗的对应数据集，然后在此模型中进行训练，预期可以达到更好的效果。

六. 参考资料

[1] 古诗数据集 https://github.com/chinese-poetry/chinese-poetry

[2] jieba分词工具 https://github.com/fxsjy/jieba

[3] He K, Zhang X, Ren S, et al. Deep residual learning for image recognition[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 770-778.

[4] Karpathy A. The unreasonable effectiveness of recurrent neural networks[J]. Andrej Karpathy blog, 2015, 21: 23.