基于neural baby talk模型的human eye

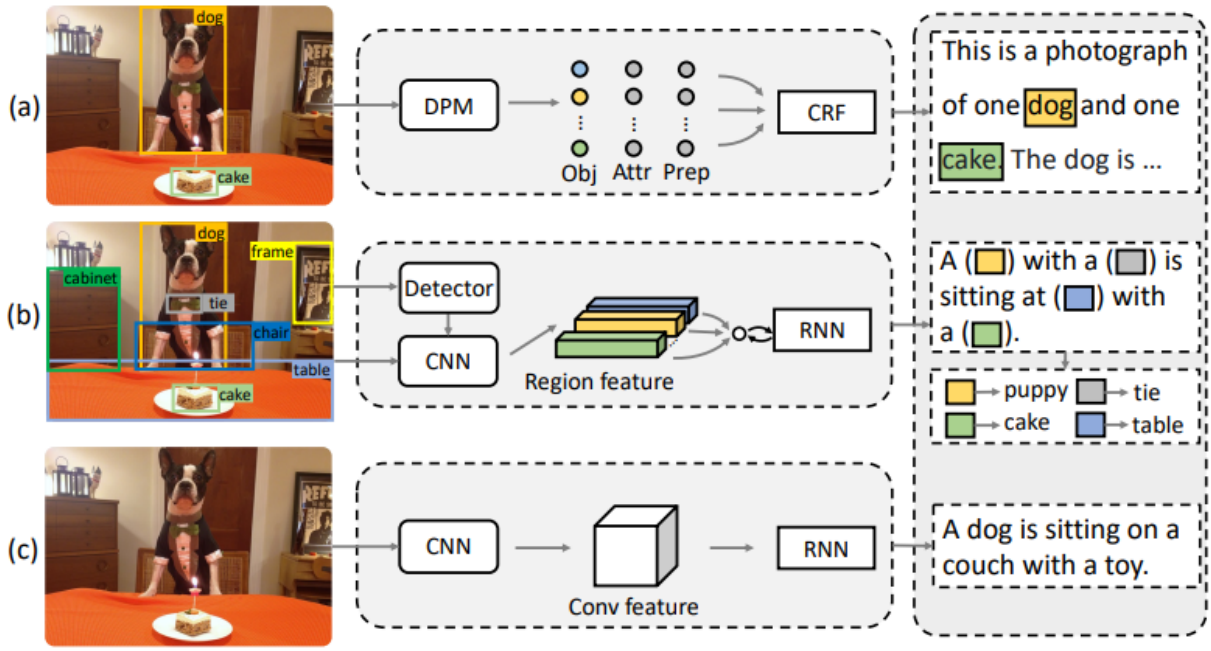
1.概述

human eye任务为对于给定的图片，生成能够描述该图片的内容的语句，要求该语句能够通过图灵测试。该类问题被归类为计算机视觉与自然语言处理领域中的image caption问题。

对于image caption问题，传统的解决方法是一个"encoder-decoder"模型，即通过CNN网络将原始图片编码为向量(encoder部分)，随后使用RNN网络将向量转变为输出语句(decoder部分)。该传统模型问题在于它事实上并不会把生成的信息与图片上的各个像素区域联系起来，并且生成出的结果与采取的训练集中的语句高度雷同，因而经常产生使人感到很别扭的语句。

由于传统模型仍有种种不足，我们小组采用了neural baby talk论文中的模型。该模型的基础结构仍为"encoder-decoder"模型，但不同的是，该模型encoder部分为使用CNN网络识别出图片上的各个region并生成每个region的特征向量，随后，仍使用RNN网络作为decoder部分，但是并非直接根据特征向量生成整个句子，而是由RNN网络生成一个带有插槽的句子，再将插槽所对应region填入该插槽。

$Figure\ 1$



如上述例子，传统的"encoder-decoder"模型直接会得到"A dog is sitting on a couch with a toy"的结果，而nerual baby talk的模型首先会得到形如"A <region-17> is sitting at a <region-

123> with a <region-3>."，随后再在每个插槽(<region-[.]>)中填入对应region的具体信息。本例中<region-17> 中具体的信息为puppy，<region-123>具体的信息为table，<region-3>中具体的信息为cake，因此最终生成的结果为"A puppy is sitting at a table with a cake."。

2.实现方法

给定图片I，模型的目标是生成基于视觉由一组单词构成的语句$y = \{y_1, ..., y_T\}$，我们用$r_I = \{r_1, ..., r_N\}$表示从图片I中N个region中提取出的N个特征向量。对于生成的描述中的每个单词，我们希望该单词能够确切地属于某一个region$r \in r_I$。遵循传统的监督学习的方式，我们按如下方法学习参数θ：

$$\theta^* = argmax_\theta \sum_{(I,y)} logp(y|I;\theta) \qquad (1)$$

使用链式法则，上式中的联合概率分布可以变换为：

$$p(y|I) = \prod_{t=1}^{T} p(y_t|y_{1:t-1}, I) \qquad (2)$$

为了消去对于模型参数的依赖，引入了变量r_t来表示y_t所在的图像中region，这样上式可以被表示为：

$$p(y_t|y_{1:t-1}, I) = p(y_t|r_t, y_{1:t-1}, I)p(r_t|y_{1:t-1}, I)) \qquad (3)$$

在该框架中，y_t可能为两种类型，一个视觉单词或者一个文本单词，分别记为y^{vis}和y^{txt}。视觉单词y^{vis}是从一个特定的图像区域r_I中生成的单词，文本单词y^{txt}是生成的描述中去掉视觉单词后剩下部分中的一个单词。它由语言模型生成，具体而言是由从语言模型中构造出的哨兵region $\overline{r}$生成（详见3.1），如在Fig.1中，"puppy"和"cake"分别由被分为"dog"和"cake"类别的region生成，是视觉单词，“with"和"sitting"并不与任何图片中的region关联，是文本单词。

由此，Eq.1可以被分解成两步，首先，最大化生成句子模板的概率，句子模板由能够生成视觉单词的region编号(插槽)和文本单词构成，如Fig1.中句子模板为"A <region-17> is sitting at a <region-123> with a <region-3>."。第二步为根据对应的region以及物体检测信息最大化视觉单词y_t^{vis}的概率，比如region中物体的分类由detector确定，在上述句子模板中，模型会在插槽中填入’puppy’，‘table’和’cake’。

接下来我们将描述如何生成带插槽的句子模板(2.1)，以及如何在插槽中填入恰当单词来获得完整的句子(2.2)。

2.1. 生成带插槽的句子模板

给定图像I,以及对应的图像描述y，首先使用预先训练好的Faster-RCNN网络来得到图像的region及region的信息。为了生成句子模板，该模型使用了传统"encoder-decoder"模型中作为decoder部分的RNN网络，在每一步中，我们根据RNN之前的隐含层h_{t-1}以及输入x_t由$h_t = RNN(x_t, h_{t-1})$计算当前隐含层h_t。在训练中，x_t使用训练集中正确的图像描述，在测试时x_t使用模型生成的y_{t-1}(teacher forcing方法)。该模型中decoder部分包含一个接受卷积特征向量组输入，带有attention机制的LSTM层。为了生成句子中视觉单词对应的插槽，模型使用带有content-based attention机制的pointer network，记$v_t \in R^{d\times 1}$是r_t由Faster R-CNN计算出的特征向量，按照如下公式计算pointing向量:

$$u_i^t = w_h^T tanh(W_v v_t + W_z h_t) \qquad (4)$$

$$P_{rI}^t = softmax(u^t) \qquad (5)$$

其中$Wv \in R^{m\times d}$，$Wz \in R^{d\times d}$和$w_h \in R^{d\times 1}$是需要被学习的参数，softmax把向量u^t标准化为所有region r_I的分布。

由于文本单词y_t^{txt}并不与任意一个region直接关联，该模型增加了一个视觉哨兵region$\overline{r}$作为与文本单词对应的region。视觉哨兵可以被认为是decoder对于已知图像信息的隐式表达。由此文本单词y_t^{txt}的概率为：

$$p(t_t^{txt}|y_{1:t-1}) = p(y_t^{txt}|\overline{r}, y_{1:t-1})p(\overline{r}|y_{1:t-1}) \qquad (6)$$

由此去掉了对I的依赖。

接下来将描述视觉哨兵是如何计算得到的，以及文本单词时如何基于视觉哨兵生成的。当作为decoder的RNN网络是LSTM时，视觉哨兵的表达式为：

$$g_t = \sigma(W_x x_t + W_h h_{t-1}) \qquad (7)$$

$$s_t = g_t \bigodot tanh(c_t) \qquad (8)$$

其中$W_x \in R^{d\times d}$，$W_h \in R^{d\times d}$。x_t是LSTM在第t步时的输入，gt是LSTM中细胞单元c_t的门，$\bigodot$表示两个矩阵对应元素相乘，σ表示sigmoid激活函数。修改Eq.5，对于包含视觉哨兵的所有region的概率为：

$$P_r^t = softmax([u^t; w_h^T tanh(W_s s_t + W_z h_t)]) \qquad (9)$$

其中$W_s \in R^{d\times d}$，$W_z \in R^{d\times d}$是参数。需要注意的是，W_z和w_h在Eq.4中是同一个参数，P_r^t是包括视觉哨兵$\overline{r}$的所有region r_I的概率分布。Eq.9中两个元素的后面一个代表着$p(\overline{r}|y_{1:t-1})$。

把隐含层h_t送入一个softmax层得到所有的文本单词内容根据前t-1步已生成的单词以及视觉哨兵region信息的概率:

$$P_{txt}^t = softmax(W_q h_t) \qquad (10)$$

其中$W_q \in R^{V\times d}$，d是隐含层的大小，V是文本单词的长度。最后将Eq.10以及Eq.9的最后一个元素($p(\overline{r}|y_{1:t-1})$)代入Eq.6得到在模板中生成视觉单词的概率。

2.2插槽的填入

为了用对应着某个region的视觉单词填充模板中的插槽，该模型使用了一个目标检测网络的输出(具体为预训练好的Faster-RCNN)。对于一个给定的region，region的类别信息可以由目标检测网络得到，但目标检测网络的输出的类别往往是单数的，指代范围非常广泛的标签，比如说"dog"，而模型希望得到的是细粒度的类别标签比如说"puppy"或者复数形式"dogs"。为了做到这种名词的变换，模型中的视觉单词y^{vis}
通过考虑以下两个因素来提高视觉单词内容的准确度，第一个因素是检测单复数，第二个因素是确定细粒度类别（如果有的话）。使用两个使用线性整流函数$f(\cdot)$的单层MLP：

$$P_b^t = softmax(Wbfb([vt; ht])) \qquad (11)$$

$$P_g^t = softmax(U^T W_g f_g([vt; ht])) \qquad (12)$$

其中$W_b \in R^{2\times d}$，$W_g \in R^{300\times d}$是权重参数，$U \in R^{300\times k}$是k个与该大类别相关的细粒度类别进行glove得到的glove vector，由此确定视觉单词y_t^{vis}的单复数形式以及细粒度类别（如果是复数，并且细粒度类别为"puppy"，则得到的视觉单词应该为"puppies"）。

3.训练准备

数据集的处理： 使用了两个数据集。Flickr30k 整体包含了来自31,783张图片的带有对应的短语或单词的275,755个bounding box。同时每个图片有5个对应的描述注释，通过使用

Stanford part-of-speech tagger选出注释中偏向名词词性的词来划分视觉单词。通过使用Stanford Lemmatization Toolbox 来获得每个词的基本形式，最终得到了2,567个不同的基础单词。

COCO包括82,783张用于训练的图片，40,504张用于检验的图片以及40775张用于测试的图片，每张图片有5个对应的描述注释，与上个数据集不同，由于COCO不带有与短语或单词一一匹配bounding box，因此使用了由论文作者提供的类别注释。

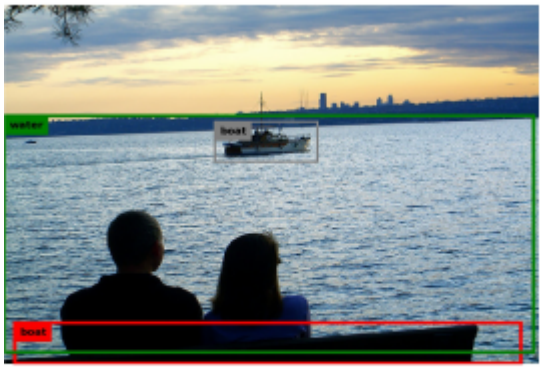
目标检测网络的预训练： 使用了现有的Faster-RCNN的开源实现。

4.训练结果

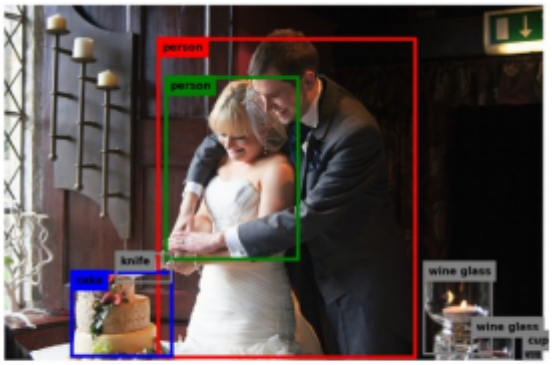
	BLUE1	BLUE4	METETOR	CIDEr	SPICE
OUR NBT	69.2	28.5	24.6	95.4	19.2
NBT	75.5	34.7	27.1	107.2	20.1

我们使用了COCO captioning evaluation toolkit来评价我们的训练结果，并与原论文的训练结果进行比对，发现与原论文的训练结果存在一定差距，并且在检查中发现对相当一部分图片的描述与预期不符，以下节选了一些较符合预期的结果进行展：

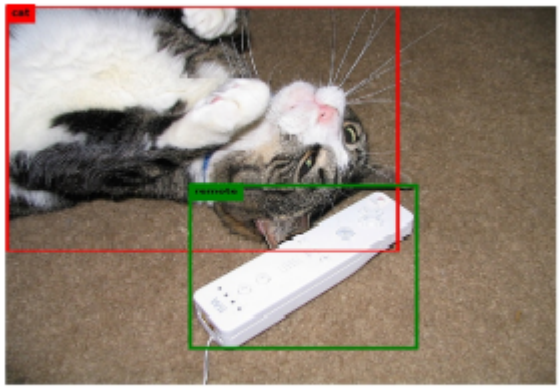
A woman standing in front of a red bus.



Two people are sitting on a boat in the water.



A bride and groom cutting a cake together.



A cat laying on the floor next to a remote control.

5.代码说明

cfgs/： 文件路径与参数信息
data/： 训练集
misc/： 模型
pooling/： faster RCNN

prepro/： 训练集文本预处理
tools/： 依赖的开源项目