**一、主要思想**

**本文的主要贡献是学习novel caption；**

1、Wg是image caption有限的训练样本中captions中词的集合，

Wc是detectionCNN标签的集合，多目标检测用MIL学习，

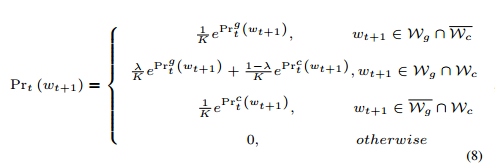
Wg和Wc有交集，也有各自独立的部分；

2、

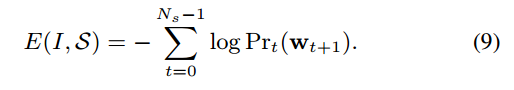
我们把每一步LSTM的输出分别用矩阵变换，得到Wg的vocabulary维度的概率和Wc的vocabulary维度的概率，如下（其中delta表示CNN分类网络的输出概率）：

C:\Users\YanJack\AppData\Local\Temp\enhtmlclip\Image(18).png

根据一个词属于Wg和Wc交并集中的哪一部分，用下式（在LSTM的输出之后增加copying layer实现）计算Wg并Wc词库中词对应的概率：



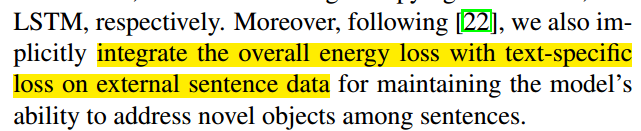
由于在交叉熵损失函数（下式）的优化过程中，每一次取image caption问题的LSTM输出词对应的ground\_truth（一定在Wg中，可能不在Wc中，换言之，一定不在Wc独立的那一部分中）的概率logP进行计算；



故训练过程中始终没有用到（8）式的第三行，也就是永远不需要计算只在Wc中出现的那一部分（novel）的词的概率，

因此并没有更新（7）式中Mc对应于novel的词的维度的那一部分参数，结论是（9）的损失函数不会work；

因此文中说：



【22】arXiv'16 Captioning images with diverse objects

即对（9）的损失函数做了改进；

**二、问题**

1、为什么计算Wc中词概率的时候对变换矩阵Mc进行了activation，而在计算Wg中词概率的时候没有这步操作？

2、detectionCNN方法的实现，包括MIL（多示例学习：如果一个包里存在至少一个正样本，则该包为正，如果一个包里所有示例都是负样本，则该包为负）；

3、text-specific loss 的实现；

4、image caption问题的end-to-end学习指什么呢？应该不是fine-tune CNN的参数吧，看到的第一篇end-to-end的论文是text-conditional；

5、SCN也能实现novel captioning，对比一下；

6、one-hot的编码顺序对学习结果有影响吗？

7、word embedding的实现？