论文笔记《Dictionary-Guided Editing Networks for paraphrase generation》

论文主要贡献：作者提出了一种基于词典指导的网络模型结构来对句子进行改写。

本文的系统框架是，首先对于原始的句子，基于PPDB的词级别和短语级别的同义词词典中检索他们的同义词，然后用一个seq2seq框架来指导哪个词应该被插入【这里说的指导其实就是采用attention机制，根据概率大小来判断】。

以前生成paraphrase的方法主要有两种：

* knowledge-based：基于知识的方法主要依赖于人造的规则和一些词典。
* Statistical machine translation(SMT)：基于统计机器翻译的方法

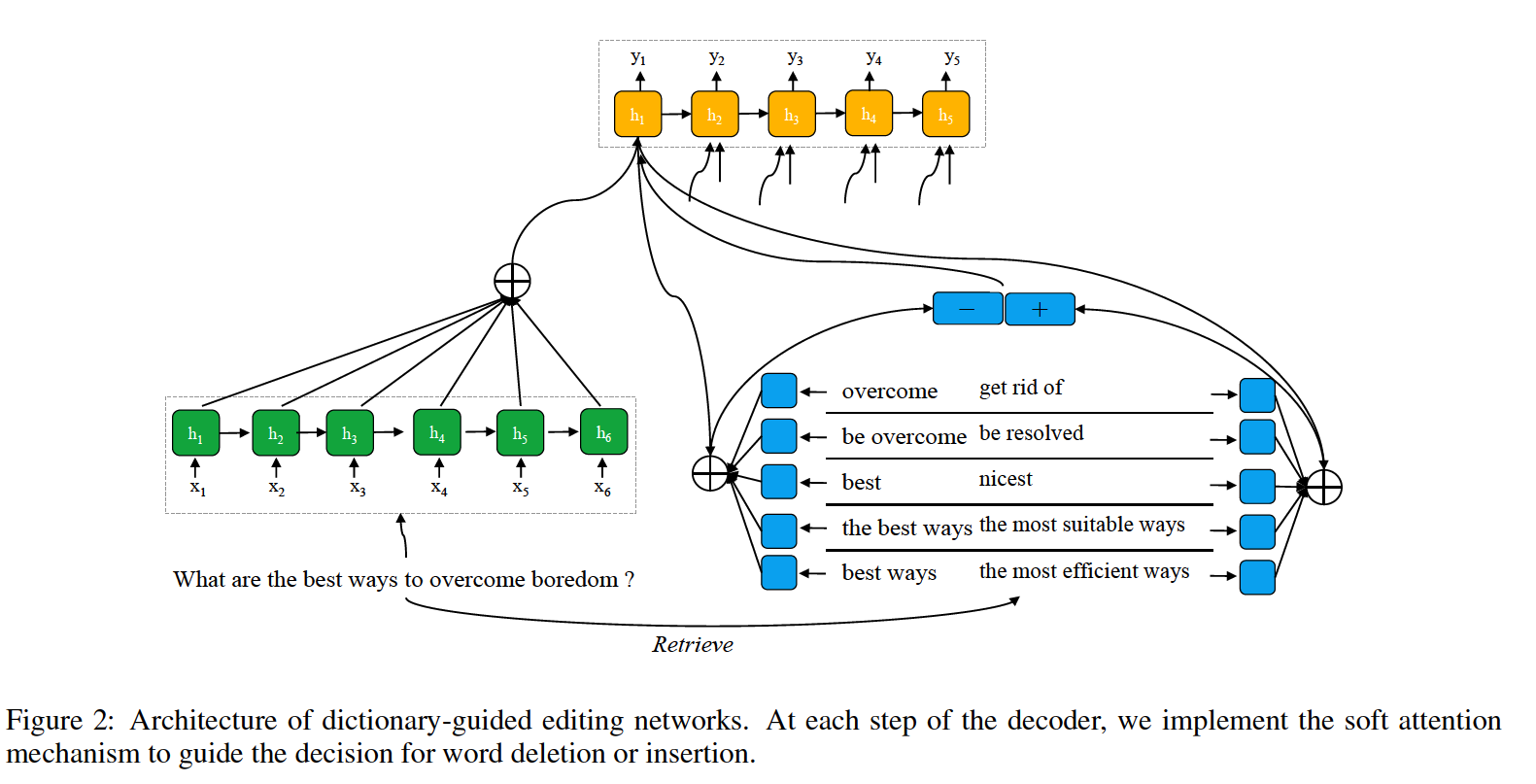
目前学术界主流的方法已经变成了基于深度学习的方法了。

**模型的流程**

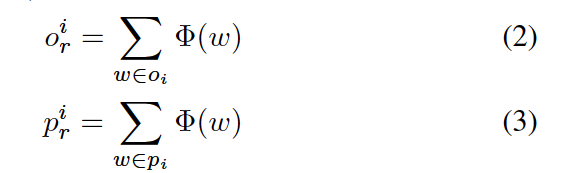
从PPDB数据集中获取一个词以及短语级别的同义词词典:，是原始词，是同义词；以及对应的paraphrase训练数据，是paraphrase pair。

作者首先检索一个同义词对，然后将这些同义词都encoder为一个向量，最后用一个神经网络模型去生成新的paraphrase。

整个流程图如下：



对于每一个同义词对，都encoder成一个向量【同义词短语，有多个单词时，将他们的每个维度的向量加起来】。



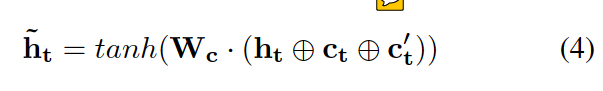
所以输入神经网络的输入有两部分：

1. 原始的句子x
2. 得到的同义词对的向量表示

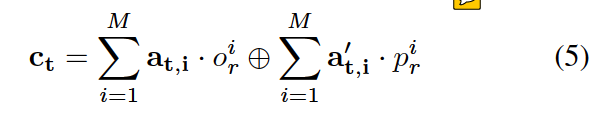
对于原始的句子x，作者采用BiRNN来获取它的融合上下文信息的新表示，并且使用attention结构获得original-side的信息。

对于同义词对，作者也采用了一个attention结构来更好的利用paraphrase dictionary的信息。

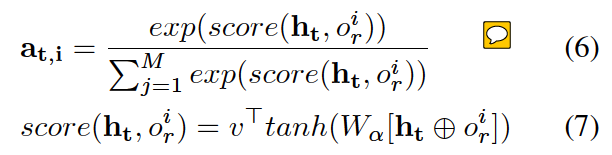
在decoder时，作者用公式4计算新的向量表示，其中ht表示BiRNN得到的隐层向量，ct是同义词对上下文向量，具体计算下面会讲到，ct'是前面source端计算attention后与source端的向量对位相乘再相加得到的表示。

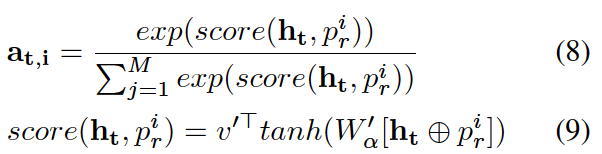


ct的计算方法如公式5：

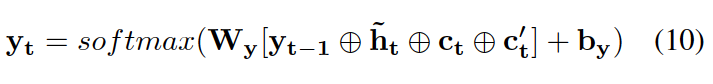


其中original同义词和paraphrase同义词也都应用了一个attention结构来获取其中的最重要信息：

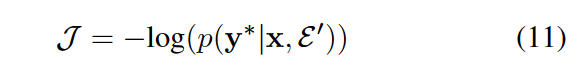




最后用上面计算的向量生成词的概率分布：



优化目标是下面的负对数似然函数：



**实验**

作者在两个数据集上进行了实验：

* MSCOS数据集：这个数据集，是对于每一个图片都有五个注释，构建paraphrase训练集时，作者从5个句子中随机选取一个target端，其余的4个作为source端，构建训练集。
* Quora数据集：这个数据集使用来识别问题是否是重复的，它包含400K个重复的问题对，还有一些不是重复的问题对【两个问题表达的含义不同】，作者采用重复的问题对作为paraphrase。

实验结果如下：

