《Convolutional Sequence to Sequence Learning》阅读笔记

周晓欢
苟日新，日日新，又日新。

论文地址：Convolutional Sequence to Sequence Learning

代码地址：facebookresearch/fairseq

这篇论文是由facebook AI团队提出，其设计了一种完全基于卷积神经网络的模型，应用于seq2seq任务中。在机器翻译任务上比以往效果更好，同时大大提高了运行速度。

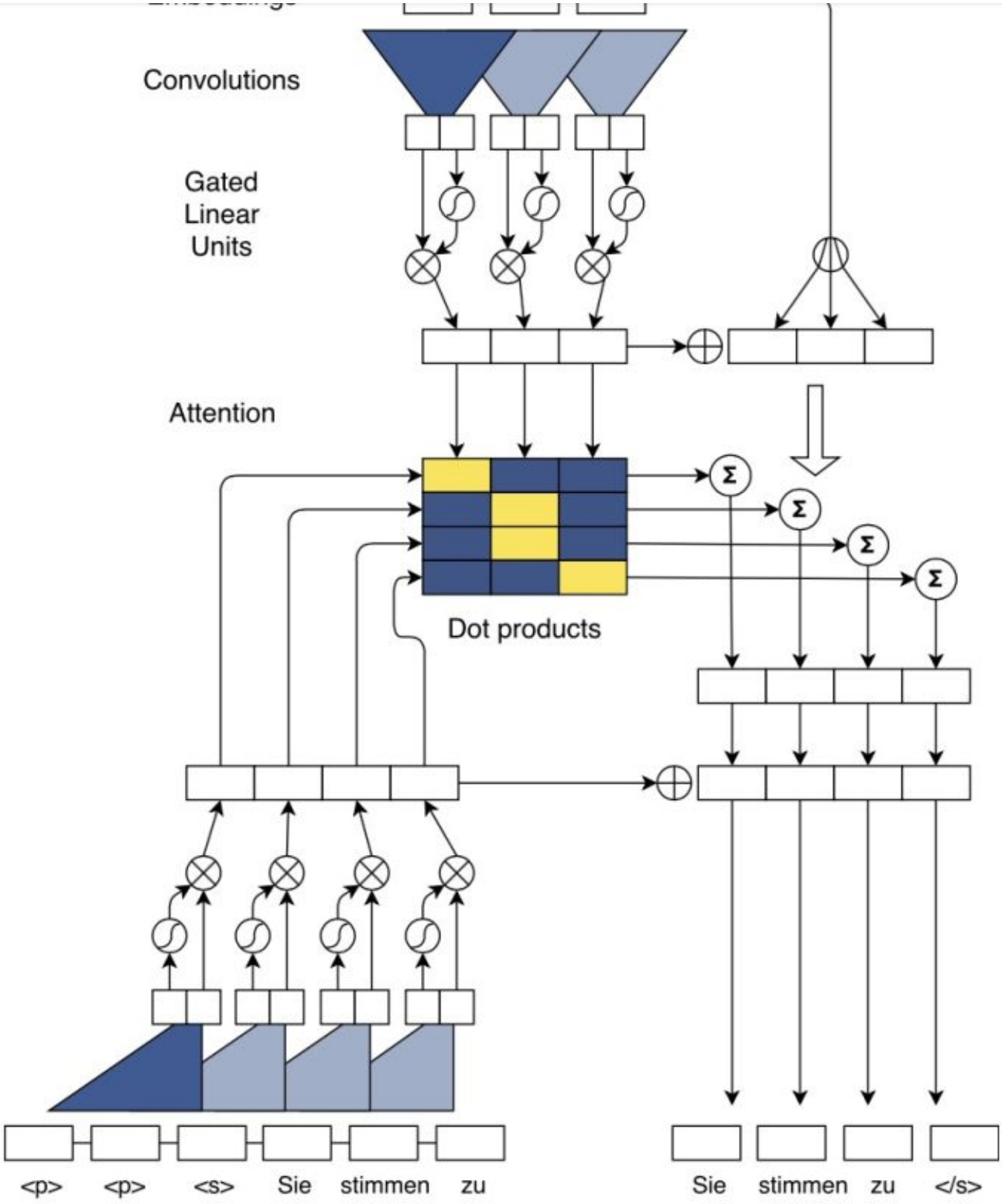
• Motivation

在以往的自然语言处理领域，包括 seq2seq 任务中，大多数都是通过RNN来实现。这是因为RNN的链式结构，能够很好地应用于处理序列信息。但是，RNN也存在着劣势：一个是由于RNN运行时是将序列的信息逐个处理，不能实现并行操作，导致运行速度慢；另一个是传统的RNN并不能很好地处理句子中的结构化信息，或者说更复杂的关系信息。

相比之下，CNN的优势就凸显出来。最重要的一点就是，CNN能够并行处理数据，计算更加高效。此外，CNN是层级结构，与循环网络建模的链结构相比，层次结构提供了一种较短的路径来捕获词之间远程的依赖关系，因此也可以更好地捕捉更复杂的关系。

• Model

整体模型结构如下图，图中表示的从英语翻译到法语的过程。该模型依旧是encoder-decoder + attention模块的大框架：encoder 和 decoder采用了相同的卷积结构，其中的非线性部分采用的是门控结构 gated linear units（GLM）；attention 部分采用的是多跳注意 multi-hop attention，也即在 decoder 的每一个卷积层都会进行 attention 操作，并将结果输入到下一层。



接下来分步讲解：

1、Position Embeddings

加入位置向量，给予模型正在处理哪一位置的信息，

位置向量：$p = (p_1, \ldots, p_n)$

最终表示向量：(输入表示向量) $e = (w_1 + p_1, \ldots, w_n + p_n)$ （输出表示向量g)

2、Convolutional Block Structure

encoder 和 decoder 都是由 l 层卷积层构成，encoder输出为 z^l ，decoder输出为 h^l 。由于卷积网络是层级结构，通过层级叠加能够得到远距离的两个词之间的关系信息。

这里把一次“卷积计算+非线性计算”看作一个单元Convolutional Block，这个单元在一个卷积层内是共享的。

卷积计算：卷积核的大小为 W^{kd*2d} ，其中d为词向量长度，k为卷积窗口大小，每次卷积生成两列d维向量 $Y = [A, B] \in R^{2d}$ 。

非线性计算：非线性部分采用的是门控结构 gated linear units（GLM）。计算公式如下：

$$v([A, B]) = A \otimes \delta(B)$$

其中， $\delta(B)$ 是门控函数，控制着网络中的信息流，即哪些能够传递到下一个神经元中。

残差连接：把输入与输出相加，输入到下一层网络中。

$$h_i^l = v(W^l[h_{i-k/2}^{l-1}, \ldots, h_{i+k/2}^{l-1}] + b^l) + h_i^{l-1}$$

输出：decoder的最后一层卷积层的最后一个单元输出经过softmax得到下一个目标词的概率。

$$p(y_{i+1}|y_1, \ldots, y_i, x) = softmax(W_o h_i^L + b_o)$$

3、Multi-step Attention

原理与传统的attention相似，attention权重由 decoder 的当前输出 h_i 和 encoder 的所有输出共同决定，利用该权重对encoder的输出进行加权，得到了表示输入句子信息的向量 c_i ， c_i 和 h_i 相加组成新的 h_i 。计算公式如下：

$$d_i^l = W_d^l h_i^l + b_d^l + g_i$$

$$a_{ij}^{l} = \frac{\exp\left(d_i^l \cdot z_t^u\right)}{\sum_{t=1}^{m} \exp\left(d_i^l \cdot z_t^u\right)}$$

$$c_i^l = \sum_{j=1}^{m} a_{ij}^l \left(z_j^u + e_j\right)$$

这里 a_{ij}^l 是权重信息，采用了向量点积的方式再进行softmax操作，这里向量点积可以通过矩阵计算，实现并行计算。

最终得到 c_i 和 h_i 相加组成新的 h_i 。如此，在每一个卷积层都会进行 attention 的操作，得到的结果输入到下一层卷积层，这就是多跳注意机制multi-hop attention。这样做的好处是使得模型在得到下一个主意时，能够考虑到之前的已经注意过的词。

- **Result**

与以往RNN模型效果做比较，明显优于RNN模型：

与以往RNN模型运行速度比较，运行速度大大提高：

- **Innovation**

将CNN成功应用于seq2seq任务中，发挥了CNN并行计算和层级结构的优势。CNN的并行计算明显提高了运行速度，同时CNN的层级结构方便模型发现句子中的结构信息。

编辑于 2017-05-23

深度学习（Deep Learning） 卷积神经网络（CNN） 机器翻译

文章被以下专栏收录

西土城的搬砖日常
机器学习，深度学习等各种人工智能分享

关注专栏

推荐阅读

《Unsupervised Machine Translation Using...

作者：Guillaume Lample, Ludovic Denoyer and Marc' Aurelio Ranzato来源： ICLR 2018 Under Review链接： link研究机构：Facebook AI Research, Sorbonne Universit´es, UPMC...

孙建东

《EFFICIENT SUMM WITH READ-AGAIN

转载请注明出处：西土城
常原文链接：EFFICIENT SUMMARIZATION WITH AGAIN AND COPY MEC
文章来源：Under review conference paper at ICL

邮递员小王 发表于西

24 条评论

切换为时间排序

写下你的评论...

czs0x55aa 1 年前

你好，请问卷积核大小kd*2d是怎么回事？感觉kd和2d都很大了，可以解释下这个卷积的过程吗

1

这里可以理解为有2d个滤波器，卷积核大小为k * d，最终生成2 * d的矩阵

2

二明 1 年前

残差连接在模型中的使用是指计算C的时候（Z + e）这个吧，这也是跟传统的循环神经网计算不同的地方，因为加e就考虑到了位置信息。当然，你说的那个我个人也认为是残差连接。

赞

周晓欢 (作者) 回复 二明 1 年前

是的，在计算c的时候也用到了

赞

abvim 1 年前

“整体模型结构如下图，图中表示的从英语翻译到法语的过程。” 图中的例子Sie stimmen zu是德语：) 小笔误

1

周晓欢 (作者) 回复 abvim 1 年前

谢谢□法语德语傻傻分不清

赞

momogary 10 个月前

你好，刚入门nlp，请问如何得到position embedding的？

赞

周晓欢 (作者) 回复 momogary 10 个月前

按照词的位置做onehot编码，得到embedding

赞

Towser 回复 周晓欢 (作者) 7 个月前

不可能 onehot 编码，维度不对。

赞

查看全部 9 条回复

kozo 7 个月前

不明白为何要position embedding，卷积不正是提取前后序列间的关系吗，既然前后序列都是知道的，还为何非得提供一个位置信息？请不吝赐教

赞

认为是基于相邻的词会有更多交互，且这些交互是有一定模式的假设。加入位置信息，能够让卷积核在计算时考虑现在正在处理那个部分的序列，这个在没有加入位置信息的时候卷积核并不清楚的，如此卷积核会进一步去考虑词之间的交互。

1

质数 6 个月前

配套上说在解码的时候同时计算四个词的 attention权重，这个怎么理解

赞

周晓欢 (作者) 回复 质数 6 个月前

不好意思，太久有些记不清了，是哪个地方呢□

赞

song xinhui 4 个月前

请问测试的时候attention是如何计算的？

赞

Canon 2 个月前

同问测试的时候attention怎么算的，因为target words是不知道的，那decoder context representation怎么获得

赞

Canon 2 个月前

GLU的作用怎么理解啊，它相当于是对d个卷积核的结果赋予不同权重，每个卷积核都接受了同样的输入信息，不同核的结果有什么区别呢

赞

sclj 1 个月前

残差连接括号()里面是表示卷积操作吗？还是简单的矩阵乘法？

赞