# 两种常见Seq2Seq的原理及公式

## 前言

我们通常使用RNN来对序列到序列问题建模，但是使用RNN建模，输出序列的长度必须和输入序列的长度相等。seq2seq框架很好地解决了这个问题。本文介绍了两种最常见的seq2seq框架。

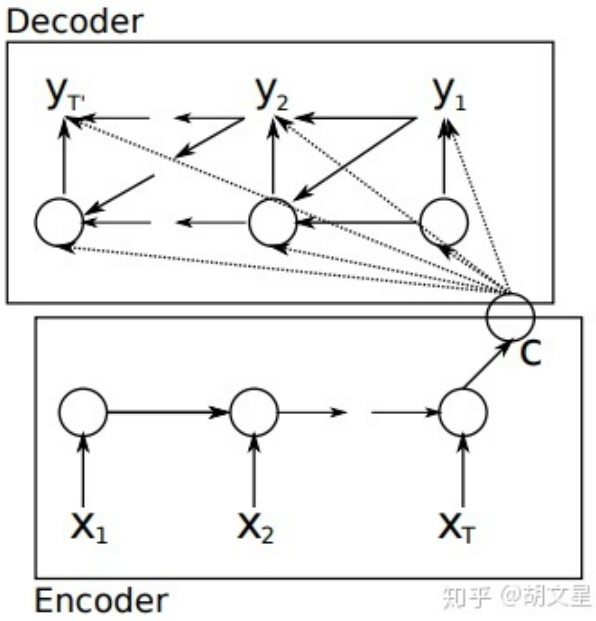
## seq2seq介绍：

seq2seq模型，全称**Sequence to sequence**，由**Encoder**和**Decoder**两个部分组成，每部分都是一个RNNCell（RNN、LSTM、GRU等）结构。Encoder将一个序列编码为一个固定长度的语义向量，Decoder将该语义向量解码为另一个序列。

**特点**：输入序列和输出序列的长度是**可变**的，输出序列长度可以**不等于**输入序列长度。
**训练**：对Encoder和Decoder进行**联合训练**，使给定输入序列的目标序列的条件概率最大化。
**应用**：seq2seq模型可以在给定输入序列的情况下生成目标序列，也可以对一对序列进行评分(以条件概率表示)。比如机器翻译、文本摘要生成、对话生成等。

## 框架1：

该框架由这篇论文提出：Cho et al.(2014) Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation。结构图如下：




这篇论文另一大贡献是提出了GRU，论文中Encoder和Decoder都是GRU。为了表达方便，这里我们假设Encoder和Decoder都为RNN，来看一下seq2seq的公式，注意： $c$ 与 $c$ 不是一个参数。

## Encoder

$$h_t = tanh(W[h_{t-1}, x_t] + b)$$
$$o_t = softmax(Vh_t + c)$$

其中 $h_t$ 是隐藏状态， $o_t$ 是输出。

Encoder输出的语义向量：

$$c = tanh(Uh_T)$$

其中 $U$ 为权重矩阵， $h_T$ 是Encoder最后的隐藏状态(记录了整个序列的信息)。

## Decoder

$$h_t = tanh(W[h_{t-1}, y_{t-1}, c] + b)$$
$$o_t = softmax(Vh_t + c)$$

接收到Encoder来的语义向量 $c$ ，首先输入一个开始信号 $y_0$ （比如为 $<START>$ ，和一个初始化的隐藏状态 $h_0$ ，接下来就按照上面的公式一直传递下去。

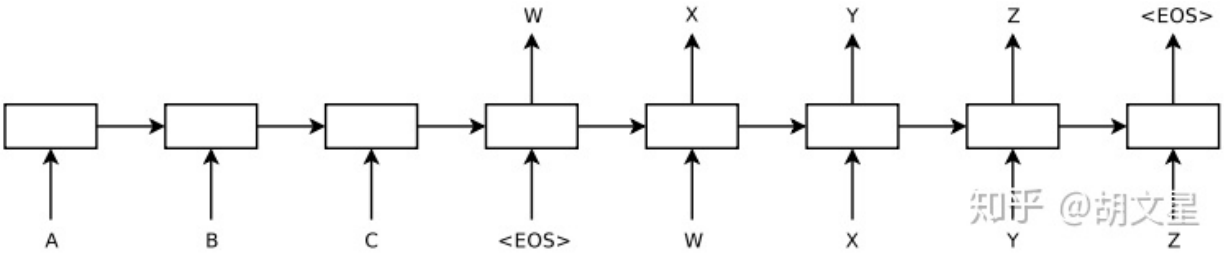
注意：语义向量 $c$ 作用于Decoder的每一时刻。

$$
\begin{aligned}
h_1 &= tanh(W[h_0, y_0, c] + b) \\
o_1 &= softmax(Vh_1 + c) \\
h_2 &= tanh(W[h_1, y_1, c] + b) \\
o_2 &= softmax(Vh_2 + c) \\
&\cdots \\
h_T &= tanh(W[h_{T-1}, y_{T-1}, c] + b) \\
o_T &= softmax(Vh_T + c)
\end{aligned}
$$

其中 $o_t$ 为每个时刻的输出，是一个向量，向量维度是词表长度，向量中的每个值是对应单词的概率。直到预测值 $< END >$ 的概率最大时，结束预测。

## 框架2：

该框架由这篇论文提出：Sutskever et al.(2014) Sequence to Sequence Learning with Neural Networks。这个框架也是最常用的一种，结构图如下：




Encoder输入序列A B C，生成语义向量 $c$ 作为Decoder的初始隐藏状态，Decoder中初始时刻输入 $< EOS >$ 作为开始标志，直至输出 $< EOS >$ 结束预测。

和框架1不同的是，该框架Encoder输出的语义向量 $c$ 直接**作为Decoder的初始隐藏状态**，并不作用于之后的时刻。

这篇论文中使用LSTM作为Encoder和Decoder，为方便描述这里用RNN作为示范，公式为：

### Encoder

$$h_t = tanh(W[h_{t-1}, x_t] + b)$$
$$o_t = softmax(Vh_t + c)$$

Encoder输出的语义向量：

$$c = h_T$$

> 论文作者发现将输入序列反转后再输入Decoder中效果会好很多，以下是由此得出的结论。
> We conclude that it is important to find a problem encoding that has the greatest number of short term dependencies, as they make the learning problem much simpler.

## Decoder

$$h_t = tanh(W[h_{t-1}, y_{t-1}] + b)$$
$$o_t = softmax(Vh_t + c)$$

其中 $h_0 = c$ 。

## 总结

seq2seq框架先用Encoder将输入序列编码成一个固定大小的语义向量，这个过程是对信息压缩的过程，不可不免地会损失很多信息，而且Decoder在解码时无法关注到输入序列的更多细节，这就引出了注意力机制，下篇文章讲Attention。

## References

[1] Cho et al.(2014) Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation
[2] Sutskever et al.(2014) Sequence to Sequence Learning with Neural Networks
[3] seq2seq学习笔记

编辑于 2019-06-26