Seq2seq

Seq2seq思想：

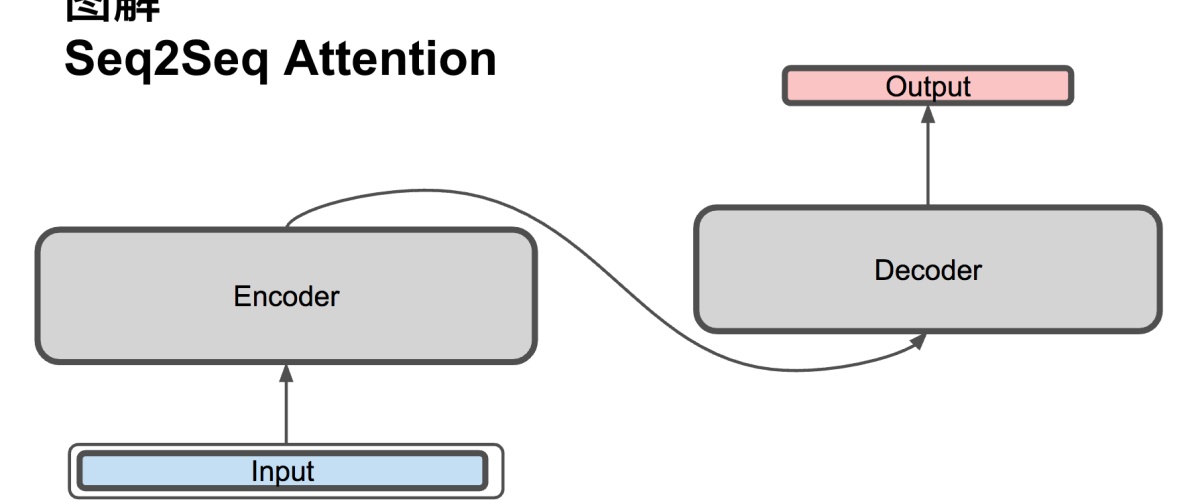
Seq2seq是encoder-decoder框架的一个算法，举例使用rnn做encoder和用rnn来做decoder。

解决的问题是：

不定长输入到不定长输出的问题。

将不定长词序列转化为定成向量表示这句话，然后解码为不定长序列。

大框架：



我理解seq2seq时碰到的几个问题：

1. 模型训练过程中到底有多少个矩阵
2. encoder是一个rnn，decoder是一个rnn
3. encoder的初始值net1怎么定，decoder的初始隐向量net1怎么定？
4. 最终decoder解码的时候输出层是一个很大的词表，如果使用softmax的话，计算时间复杂度是不是很高？没用优化算法吗？

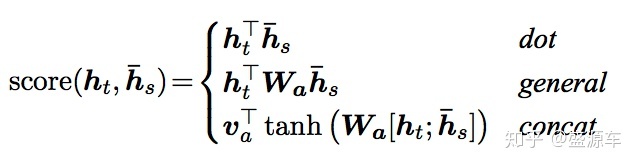
下面是整体流程：

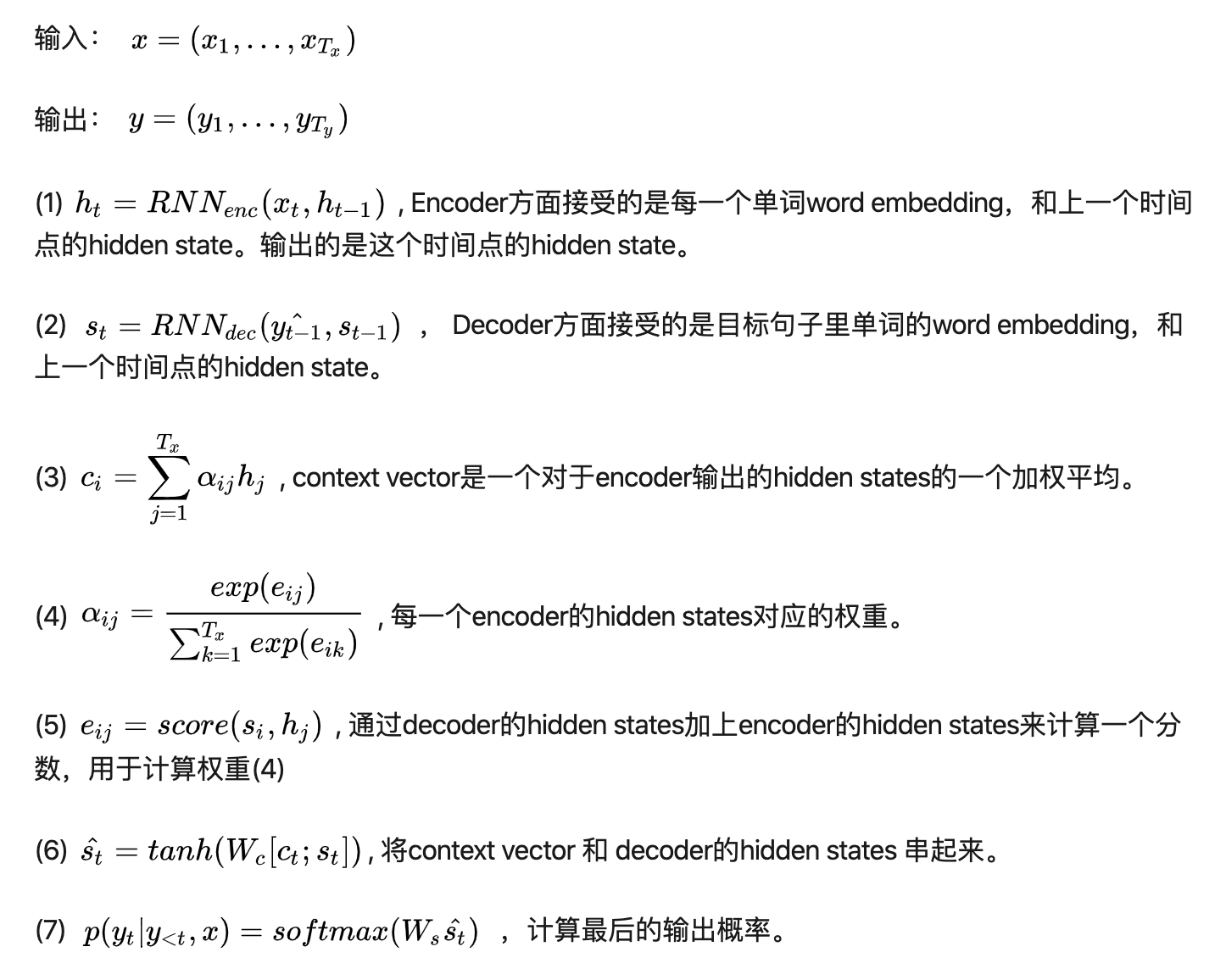
理解的时候比较费力的点：

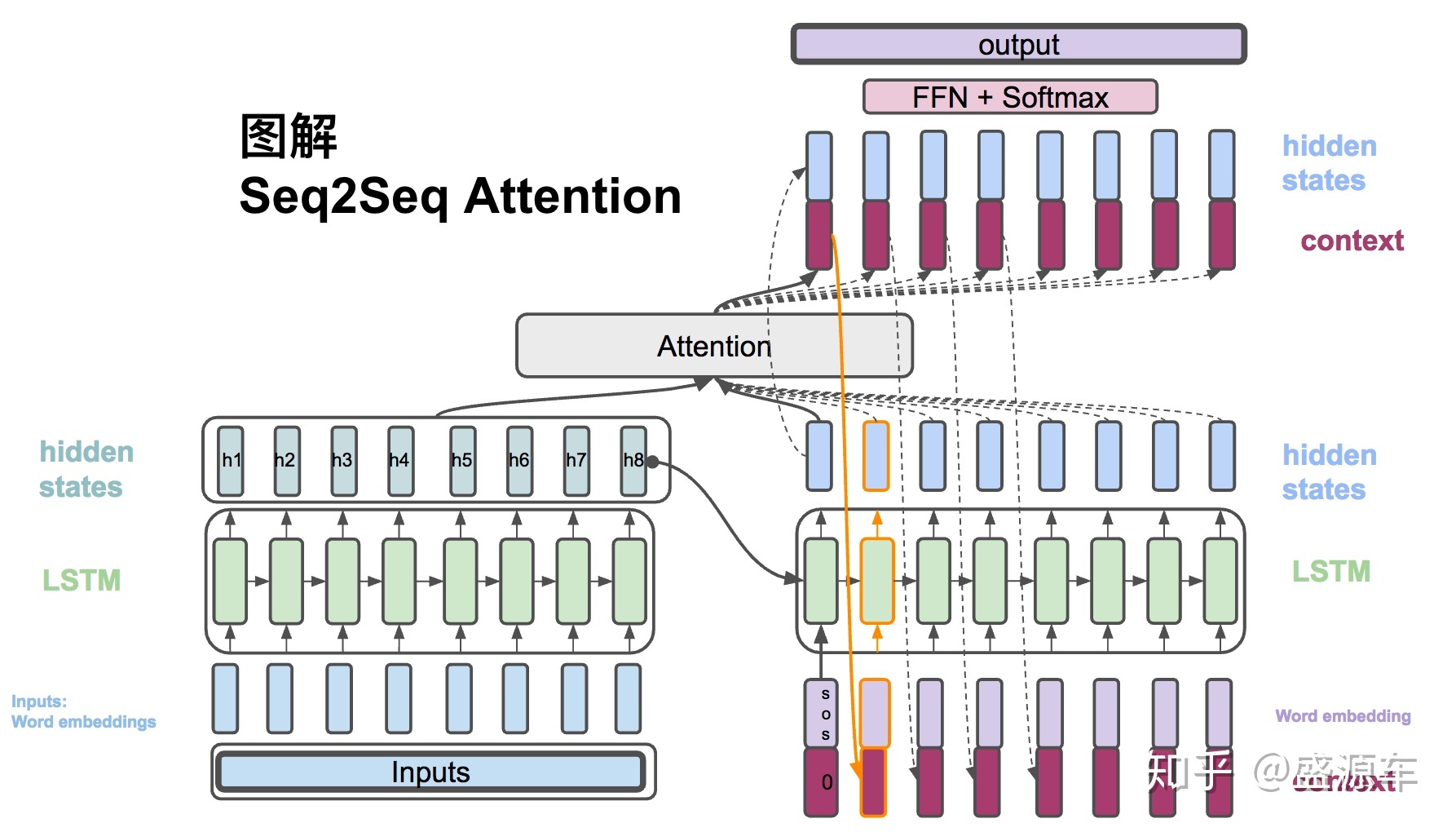
1. attention怎么计算？思路如下很清晰

attention的计算方法有多种（在求score那块）

思路：简单的点积方法，或者是去学习一个矩阵（第二种方法）然后计算score，或者学习更多的参数来计算score







详解上面的图：

1. encoder的rnn和decoder的rnn是两个独立的rnn，他们俩的参数可以看成的独立的
2. 两个rnn通过attention间接的连接在一起，attention之后得到的ct和st做拼接，其中st由s(t-1)和相对应的前一个y(t-1)预测值通过一个rnn cell获得，这一点跟encoder不一样，encoder使用的是xt。而y(t-1)预测值是st-1和c(t-1)拼接之后经过全联接，然后softmax分类而成。

Seq2seq比我想象的要复杂，主要在解码那一块。

Seq2seq经常会使用beam search做优化。因为如果第一个预测的词预测错了或者中间的某一个词预测错了，后面的词就可能会全错。

Beam search的思想就是每次预测topk个词。

如果每次都预测topk个词。

Beam search在模型训练中不需要使用，因为在模型训练中，预测下一个词的时候会用正确的词进行预测。

Beam search 在seq2seq中的解码中的应用：

1. 在解码第一个词的时候，softmax选择topk的词。
2. 在解码第二个词的时候，这topk的每个词都会拿来预测第二个词，每个词都产生topk个候选词。这个时候在第二个词的时候就有topk^2个词作为候选。
3. 减枝思想。从topk^2个词中选择topk的值作为第二个词的候选词
4. 重复2）3）即可。

5）在解码到最后一个词的时候选择top1