

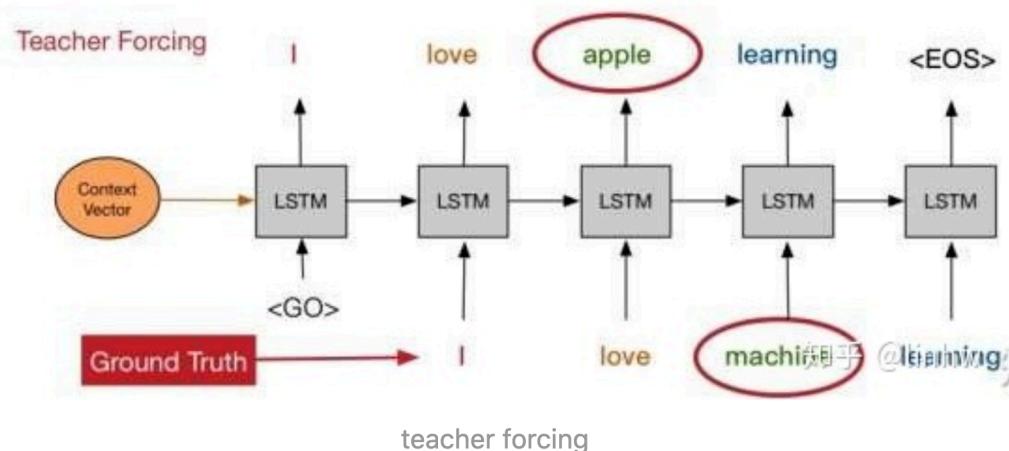
这是我学习CS224N的一些笔记和问题

以下是一些我做的笔记：

1. seq2seq在训练时与测试时有什么不一样？

seq2seq训练时，decoder的输入是ground truth，而测试时的输入是前一个decoder的输出。

Teacher Forcing可能会导致exposure bias。



在机器翻译的模型中通常使用的是seq2seq模型。在seq2seq模型的解码端中，当前词是根据前一个词来生成的，但是在训练时使用的是Teacher Forcing，前一个词是从ground truth中得到的，而在测试时，前一个词是模型自己生成的，这就使得在训练和测试时预测出的单词实际上从不同的分布中得到的，这就是exposure bias。由于exposure bias的存在，在测试时，如果某一步出现错误，那么错误就会一直累积（因为训练时前一个单词总是正确的），最终导致生成不正确的文本。

除此之外作者认为在机器翻译的过程中还有另一个问题：overcorrection。即在机器翻译中，损失函数通常是交叉熵，交叉熵函数会严格匹配预测的输出和ground truth是否一致，如果预测的词和ground truth中的词不同，尽管这个翻译是合理的，但也会被交叉熵纠正，这降低了翻译结果的多样性。

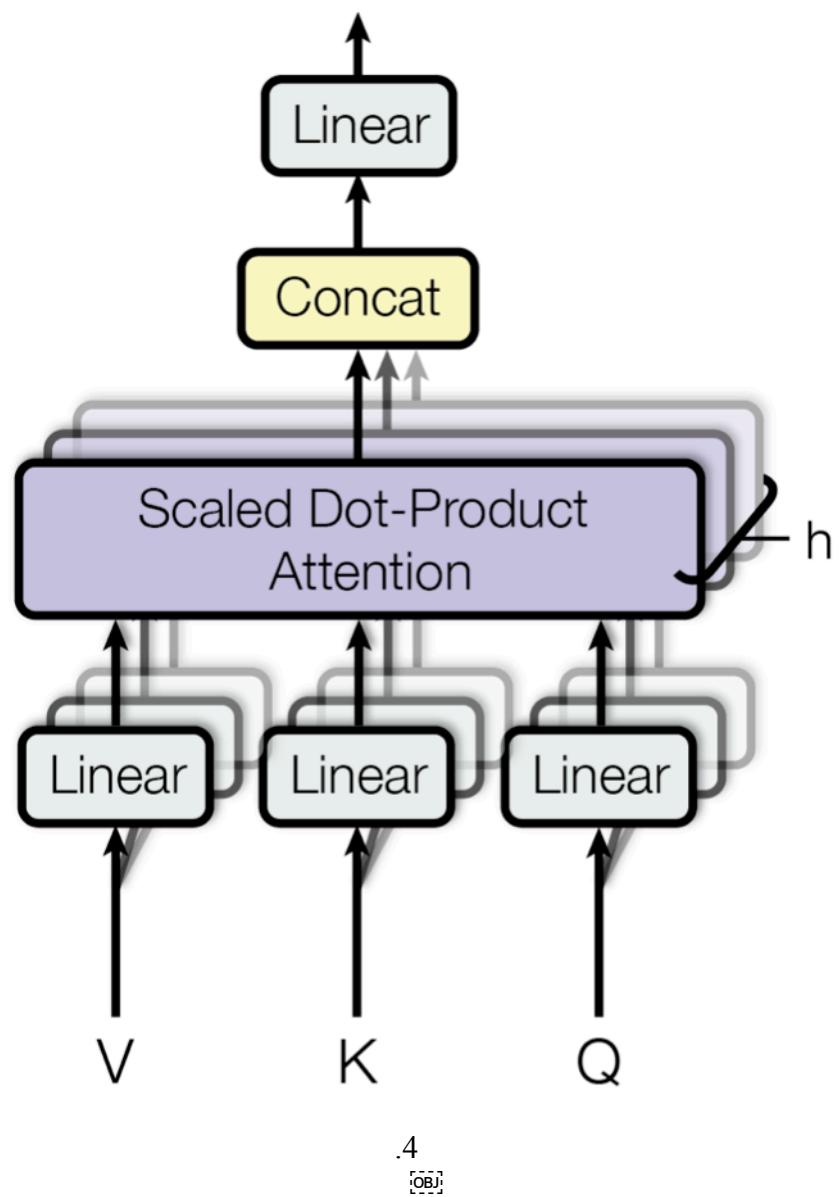
2. transformer中encoder和decoder的区别？

2.1 encoder中有两层sub-layer (multi-head self-attention和fully connected feed-forward)，decoder中有三层sub-layer (Masked Multi-head Attention、Encoder-Decoder Attention和Feed Forward)。每个sub-layer都加了Add&Norm (residual connection和layer Normalization)

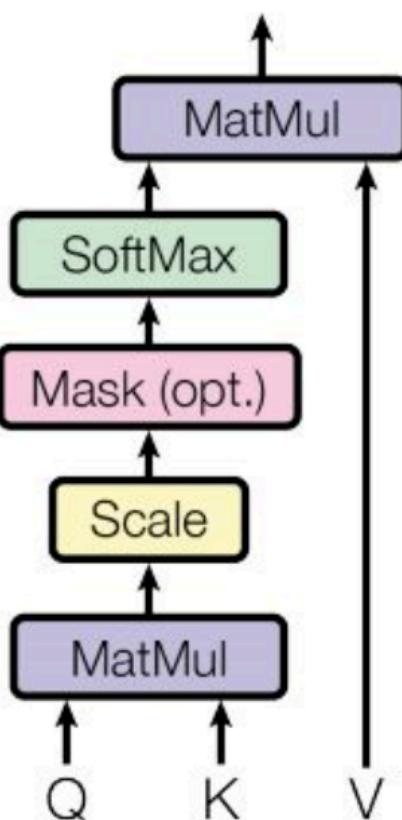
2.2 encoder中是并行计算的，而decoder也得按顺序得出一个一个的结果。

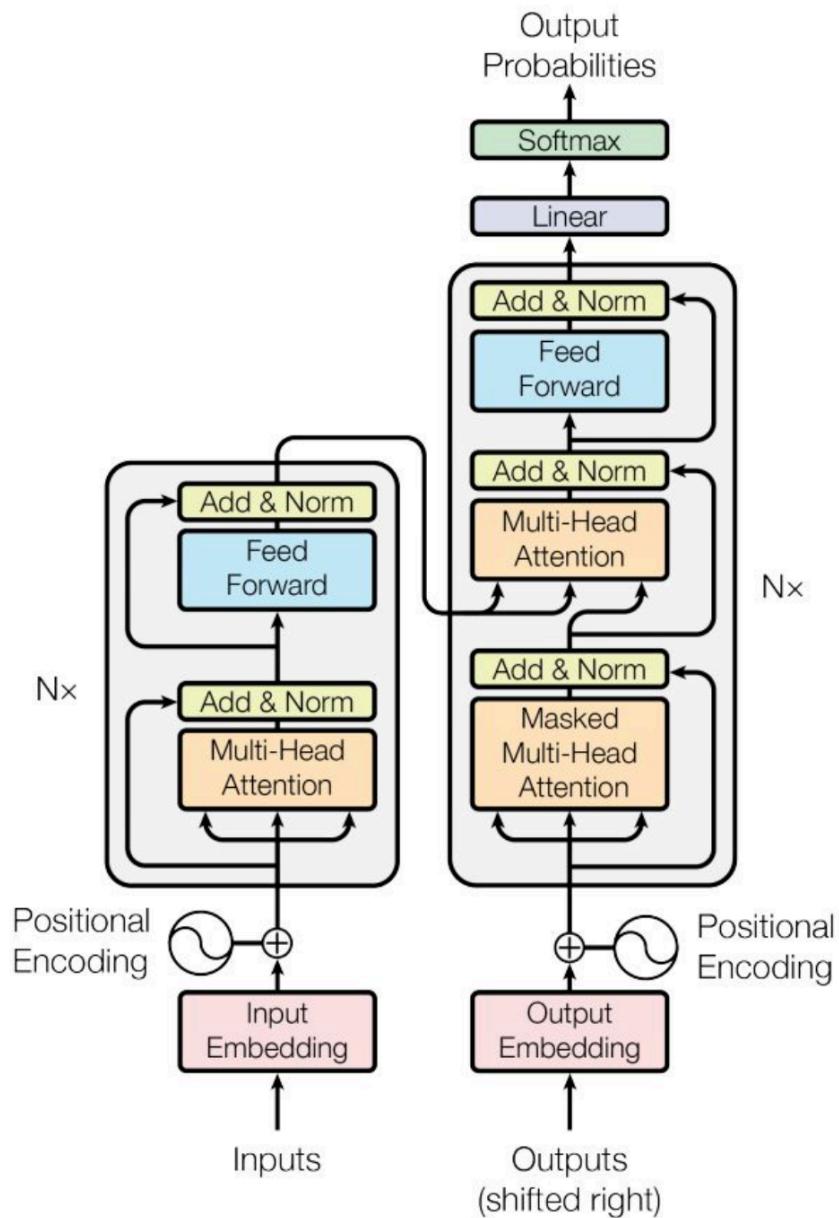
2.3 encoder-decoder attention的输入：encoder的输出和前一个decoder的输出

Multi-Head Attention



Scaled Dot-Product Attention





3. beam search的思想?

在Beam Search中只有一个参数B，叫做beam width(集束宽)，用来表示每一次挑选top B的结果。

4. word embedding如何解决一词多义的问题?

可以赋予文章中的同一个word不同位置的词一个权重，最终输出的word embedding是一个weighted sum

5. 讲一讲word2vec的negative sampling和hierarchical softmax?

本质是word2vec高效训练的计算框架。

5.1 负采样：选取k个负样本（错误分类的样本），通过k个负样本，将模型复杂度由 $O(V)$ 降至 $O(K)$

5.2 层序softmax：讲单词以Huffman树的形式组织起来，频率最高的所对应的路径最短，模型复杂度降至log倍数

6. LayerNormalization和BatchNormalization的区别？

6.1 BatchNorm是对同一个batch里的不同data的同样的dimension做 normalization，我们希望整个batch里面同一dim的mean=0, variance=1。

6.2 LayerNorm是不用考虑Batch的，给一笔data，我们希望各个不同dim的 mean=0, variance=1。LayerNorm一般会搭配RNN一起使用，

7. 做Attention的方式除了Inner Product还有别的吗，这些有什么不同？

7.1 Soft Attention：

传统的Attention Mechanism就是Soft Attention, 即通过确定性的得分计算来得到attended之后的编码隐状态。Soft Attention是参数化的 (Parameterization)，因此可导，可以被嵌入到模型中去，直接训练。梯度可以经过Attention Mechanism模块，反向传播到模型其他部分。

也有称作TOP-down Attention。

7.2 Hard Attention：

相反，Hard Attention是一个随机的过程。Hard Attention不会选择整个 encoder的隐层输出做为其输入，Hard Attention会依概率 S_i 来采样输入端的隐

状态一部分来进行计算，而不是整个encoder的隐状态。为了实现梯度的反向传播，需要采用蒙特卡洛采样的方法来估计模块的梯度。

两种Attention Mechanism都有各自的优势，但目前更多的研究和应用还是更倾向于使用Soft Attention，因为其可以直接求导，进行梯度反向传播。

8. BERT里面训练LM的随机替换为什么是结果更好？

基于 interpolation 的 LM 平滑

9. 为什么需要position embedding？

因为每一个input都要做一次self-attention，而self-attention的softmax输出考虑了全部的input，即考虑了全部的信息，那这样对于所有的attention输出，并没有考虑到具体的输入顺序，可以说第一个和最合一个最起来是一样的，所以需要PE来提取输入句子中单词的顺序信息。

10. self-attention的流程？

每一个input embedding分别乘以Q、K、V三个Matrix得到qi、ki、vi，然后拿每个query qi对所有的key ki做attention(一般为inner product) 得到一个attention score，然后通过softmax层，得到score-hat，最后每一个与对应的value vi进行矩阵相乘，得到b1，即self-attention该层的输出。注意的是b1-b4可以parallel地计算出来。

self-attention解决了并行计算、长距离依赖的问题

11. 什么是迁移学习

12. 什么是ELMo？

ELMo: Embeddings from Language Models。

在EMLo中，他们使用的是一个双向的LSTM语言模型，由一个前向和一个后向语言模型构成，目标函数就是取这两个方向语言模型的最大似然。

在预训练好这个语言模型之后，ELMo就是根据下面的公式来用作词表示，其实也就是把这个双向语言模型的每一中间层进行一个求和。最简单的也可以使用最高层的表示来作为ELMo。

总结一下，不像传统的词向量，每一个词只对应一个词向量，ELMo利用预训练好的双向语言模型，然后根据具体输入从该语言模型中可以得到上下文依赖的当前词表示（对于不同上下文的同一个词的表示是不一样的），再当成特征加入到具体的NLP有监督模型里。

13. 什么是GPT?

目标是学习一个通用的表示，能够在大量任务上进行应用。

然后再具体NLP任务有监督微调时，与ELMo当成特征的做法不同，OpenAI GPT不需要再重新对任务构建新的模型结构，而是直接在transformer这个语言模型上的最后一层接上softmax作为任务输出层，然后再对这整个模型进行微调。他们额外发现，如果使用语言模型作为辅助任务，能够提升有监督模型的泛化能力，并且能够加速收敛。

14. 讲一讲BERT?

BERT这篇论文把预训练语言表示方法分为了基于特征的方法（代表ELMo）和基于微调的方法（代表OpenAI GPT）。而目前这两种方法在预训练时都是使用单向的语言模型来学习语言表示。

BERT最大输入长度为512。

15. fine-tune怎么做?

我们假设在Resnet101后面加上一个全连接层，然后我们锁住前面Resnet的参数，不参加梯度更新，然后只更新最后一个全连接层的参数。当全连接层的

loss足够小的时候，再释放所有的参数一起训练。这样Resnet的参数也会微微调整，这就是finetune；

16. SubWord Model是什么？

BPE：Byte Pair Encoding，即寻找经常出现在一起的Byte对，合并成新的Byte Pair加入到词汇库。

课程在这里介绍了介于word-level和char-level之间的Sub-word models，主要一般有两种结构，一种是仍采用和word-level相同的结构，只不过采用更小的单元word pieces来代替单词；另一种是hybrid architectures，主要部分依然是基于word，但是其他的一些部分用characters。

16.1 Hybrid architecture的核心思想：大部分时候都使用word-level的模型来做translate，只有在遇到rare or unseen的words的时候才会使用character-level的模型协助。这种做法产生了非常好的效果。

混合模型即两种方式并存的模型，在正常处理时采用word-level的模型，当出现奇怪的词的后，使用char-level级的模型。

17. 什么是high-way Network？与Residual Network的区别？

18. 11.16-11.22这周看了什么？

18.1 指代消解（Coreference Resolution）

1、什么是指代消解？

将代表同一实体（Entity）的不同指称（Mention）划分到一个等价集合（指代链，Coreference Chain）的过程称为指代消解。

2、指代消解一般分为两个步骤：

Detect所有指称 和 Cluster指称

3、指称*Detection*做法：

可以训练一个分类器过滤掉spurious mentions；更为常见的是，保持所有的mentions作为‘candidate mentions’，在指代系统运行完后，丢弃所有的单个mention（即没有被标记为与其他东西共同引用的）

4、照应词/回指*anaphora*与指代*coreference*有区别：

- 1、不是所有的名词短语都有指代（No dancer twisted her knee）
- 2、不是所有的照应关系都有指代（concert -> ticket），这叫桥接回指
bridging anaphora

5、四种*coreference Models*：

1、*Ruled-Based*

①Hobbs算法，这是手工写的规则集，在句法树上运行，类似于“如果...则跳转到第几步...否则...”，该算法根据英文语言直觉上的直觉编写，用来寻找coreference；

②基于知识库的指代消解：利用外部世界知识编码成共指问题，e.g. 从水壶倒水直到水杯满了/空了。但是效果不太好

2、*Mention Pair*

①共指对：将所有的指代词（短语）与所有被指代的词（短语）视作一系列pair，对每个pair二分类决策成立与否。

②共指连接具有传递性：即使没有存在link的两者也会由于传递性，处在同一个聚类中。**BUT**，这是危险的，因为万一有个错了，那么有可能所有的東西都处在同一个cluster当中

③另外，在Model中，我们为每个mention预测所有的pair的做法不太合理，因为很多时候，每个mention只有一个清晰的先行词，更好的做法是，为每一个mention只预测一个先行词。

3、*Mention Ranking*

①根据模型把得分最高的先行词分配给每个mention

②虚拟的NA mention 允许模型拒绝将当前的mention与任何内容联系起来
(singleton or first mention)

③对于mi、mj是共指的情况，我们希望能够最大化概率

④那如何计算概率？

A、非神经统计的分类器

利用Features：语义相容性、语法约束、人、数字、排比等

B、简单的神经网络（FFNN）

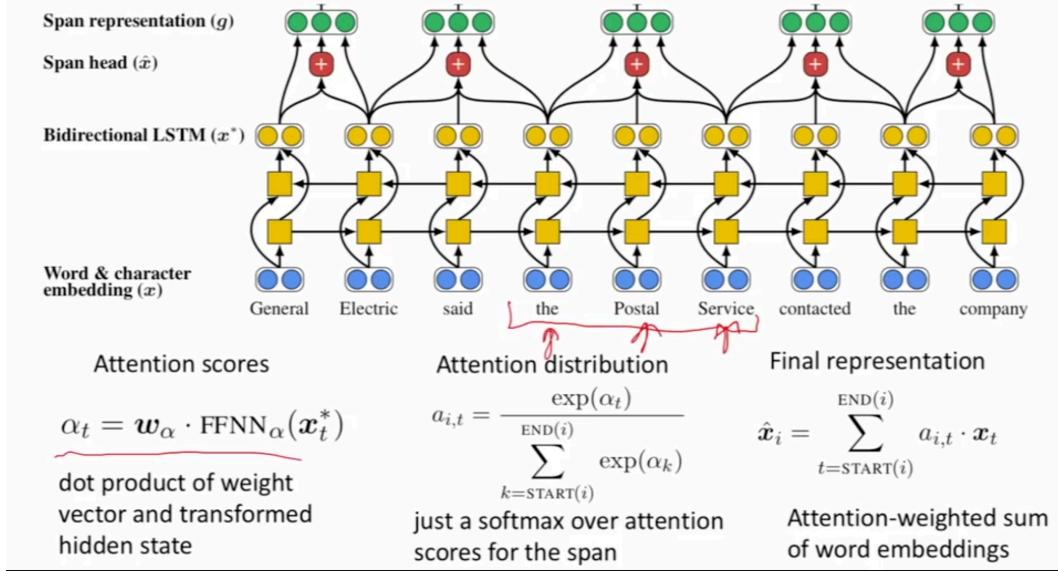
Input：先行词与mention的embedding和features（距离、文档载体）

C、更高级的利用LSTMs、attention的模型

即End-to-End模型，2017年的一篇论文

End-to-end Model

- \hat{x}_i is an attention-weighted average of the word embeddings in the span



将每段文本*i*从start(*i*)到end(*i*)表示为一个向量，最后为每个span对打分来决定他们是不是共指mentions

span representation: $gi = [x\text{-start}, x\text{-end}, x\text{-attention}, x\text{-other}]$

4、Clustering

①自下而上的bottom-up。一开始，每个mention在单独的集群中，每一步合并两个集群，使用模型来打分哪些聚类合并是好的。mention-pair 难做的，cluster-pair可能会变得容易。

②google提出了个聚类模型框架：首先为每个mention pair生成一个向量，接着pooling操作作用在这个mention-pair矩阵上，得到一个cluster-pair对的表示，然后通过weight matrix与cluster merge的dot product来计算得分

6、Coreference Evaluation

对多不同的评价指标：MUC、B-cubed、LEA

例如 B-cubed：

18.2 对于每个mention，计算其召回率和准确率

18.2.1 然后平均每个个体的准确率和召回率

18.3 Multitask Learning

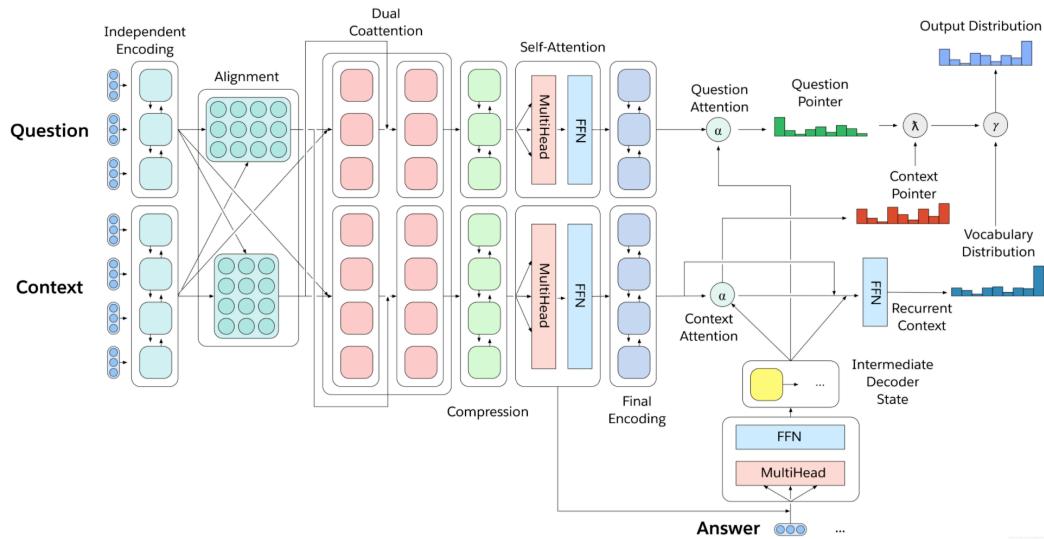
I、多任务学习

定义：基于共享表示（shared representation），把多个相关的任务放在一起学习。

优点：更容易适应新的任务、简化部署到生产的时间、。。。

2、decaNLP

①为了有效地在所有decaNLP中进行多任务处理，我们引入了MQAN，一个
多任务问题回答网络(Multi QA Network)，它没有任何针对特定任务的参数
和模块。把10项不同的任务都写成QA的形式，进行训练和测试



②简单地说，MQAN采用一个问题和一个上下文背景文档，用BiLSTM编
码，使用**Dual Coattention**对两个序列的条件进行表示，用另两个BiLSTM压
缩所有这些信息，使其能够更高层进行计算，用**self-attention**的方式来收集

这种长距离依赖关系，然后使用两个BiLSTM对问题和背景环境的进行最终的表示。多指针生成器解码器着重于问题、上下文以及先前输出象征来决定是否从问题中复制，还是从上下文复制，或者从有限的词汇表中生成。

③训练策略：Fully Joint; Anti-curriculum Pre-training由难到易

④什么是Coattention：协同注意力Co-Attention是注意力机制的一种变体，以机器阅读理解为例，注意力机制就很像我们人在做阅读理解时所使用的一种技巧——带着问题去阅读，先看问题，再去文本中有目标地阅读以寻找答案。而机器阅读理解则是通过结合问题和文本段落二者的信息，生成一个关于文本段落各部分的注意力权重，对文本信息进行加权，该注意力机制可以帮助我们更好的去捕捉文本段落中和问题相关的信息。**Co-Attention**则是一种双向的注意力，不仅我们要给阅读的文本段落生成一个注意力权重，还要给问句也生成一个注意力权重。

⑤DCN模型：Dynamic Coattention Network，动态迭代Dynamic iteration也是NLP领域一种比较前沿的技术思想，其主要思想在于仿照人类在考量问题时，需要反复思考。对于模型输出的结果，我们不直接将它作为最终的结果，而是将它继续输入到模型中作为参考，迭代出新一轮的输出，经过多次迭代，直到输出不再变化或超过迭代次数阈值。

⑥Anti-curriculum：

18.4 Constituency Parsing, Tree Recursive Neural Networks (TreeRNNs)

为了弄清楚更大的短语的含义，而不仅仅是词向量。人们通过较小元素的予以成分来解释较大文本单元的意义。递归对于描述语言是很自然的，就像树的结构一样，一层一层往下分解。

我们怎样表示更长的短语意思，并将他们映射到相同的向量空间？

基于组合规则，学习解析树以及组合向量的表示

Recursive vs. Recurrent | 递归 vs. 循环

递归神经网络需要树的结构来处理信息；循环神经网络以词的序列来处理信息，不能在没有上下文的情况下捕捉短语

递归合并过程：

自左向右重复遍历，先计算词向量两两合并的分数，然后挑选分数最高的合并（greedy），依此循环，直到最终合并为根节点。

递归的矩阵向量空间中的组合 Recursive Matrix-Vector Spaces

①提出背景：“very good”中，“very”是没有意义的，只是用来修饰“good”，那么我们在做运算的时候，可以将very视为作用在good矩阵上的向量。

②每个单词都拥有一个向量意义和一个矩阵意义，运算可以捕获语义，即其中哪一个词修饰了另外一个词的含义（MV-RNN）

③MV-RNN主要不足：否定积极（改变一个词不足以改变整个句子）；双重否定

递归神经张量网络 Recursive Neutral Tensor Network

①比MV-RNN参数更少

②不仅仅是将两个表示单词的向量相互作用，还在中间插入了一个矩阵（这是张量的切片），以双线性的方式作用，然后输出一个分数