21. Pointer-Generator Networks（指针生成网络）

1．文本摘要生成

文本摘要旨在将文本或文本集合转换为包含关键信息的简短摘要。按照输出类型可分为抽取式摘要和生成式摘要。抽取式摘要从源文档中抽取关键句和关键词组成摘要，摘要全部来源于原文。生成式摘要根据原文，允许生成新的词语、原文本中没有的短语来组成摘要。指针生成网络属于生成式模型。

2．仅用Neural sequence-to-sequence模型可以实现生成式摘要，但存在两个问题：

- 可能不准确地再现细节, 无法处理词汇不足（OOV）单词;
- 倾向于重复自己。

3．指针生成网络（Pointer-Generator-Network）从两个方面进行了改进：

- 该网络通过指向（pointer）从源文本中复制单词，有助于准确地复制信息，同时保留通过生成器产生新单词的能力;
- 使用coverage机制来跟踪已总结的内容，防止重复。

4．Baseline sequence-to-sequence

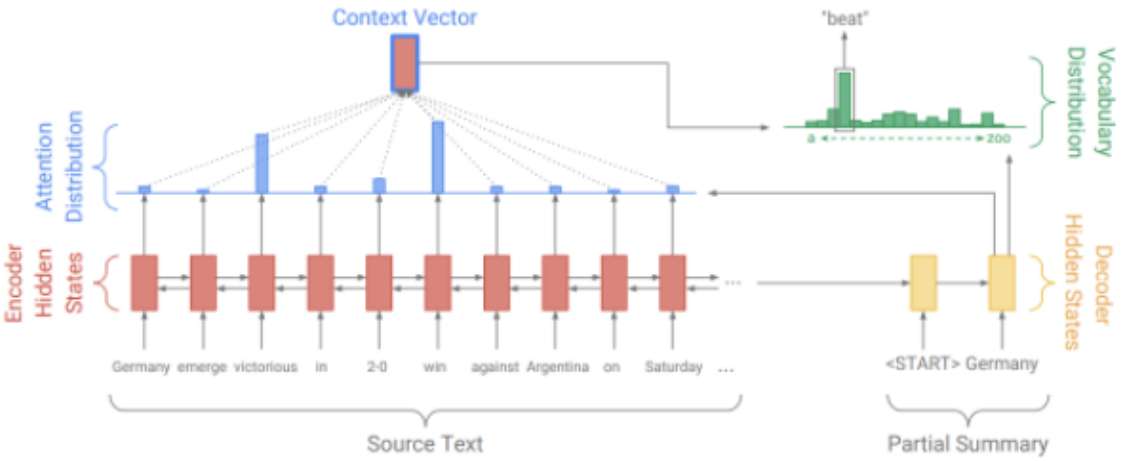


Figure 2: Baseline sequence-to-sequence model with attention. The model may attend to relevant words in the source text to generate novel words, e.g., to produce the novel word *beat* in the abstractive summary *Germany* ***beat*** *Argentina 2-0* the model may attend to the words *victorious* and *win* in the source text.

该模型可以关注原文本中的相关单词以生成新单词进行概括。比如：模型可能注意到原文中的"victorious"和" win"这个两个单词，在摘要"Germany beat Argentina 2-0"中生成了新的单词beat 。

Seq2Seq的模型结构是经典的Encoder-Decoder模型，即先用Encoder将原文本编码成一个中间层的隐藏状态，然后用Decoder来将该隐藏状态解码成为另一个文本。Baseline Seq2Seq在Encoder端是一个双向的LSTM，这个双向的LSTM可以捕捉原文本的长距离依赖关系以及位置信息，编码时词嵌入经过双向LSTM后得到编码状态 hi 。在Decoder端，解码器是一个单向的LSTM，训练阶段时参考摘要词依次输入(测试阶段时是上一步的生成词)，在时间步 t得到解码状态 st 。使用hi和st得到该时间步原文第 i个词注意力权重。

$$e_i^t = v^T tanh(W_h h_i + W_s s_t + b_{attn})$$

$$a^t = softmax(e^t)$$

得到的注意力权重和 h_i 加权求和得到重要的上下文向量 $h_t^*(contextvector)$：

$$h_t^* = \sum_i a_i^t h_i$$

h_t^* 可以看成是该时间步通读了原文的固定尺寸的表征。然后将 s_t 和 h_t^* 经过两层线性层得到单词表分布 P_{vocab}：

$$P_{vocab} = softmax(V'(V[s_t, h_t^*] + b) + b')$$

其中 $[s_t, h_t^*]$ 是拼接。这样再通过 $softmax$ 得到了一个概率分布，就可以预测需要生成的词：

$$P(w) = P_{vocab}(w)$$

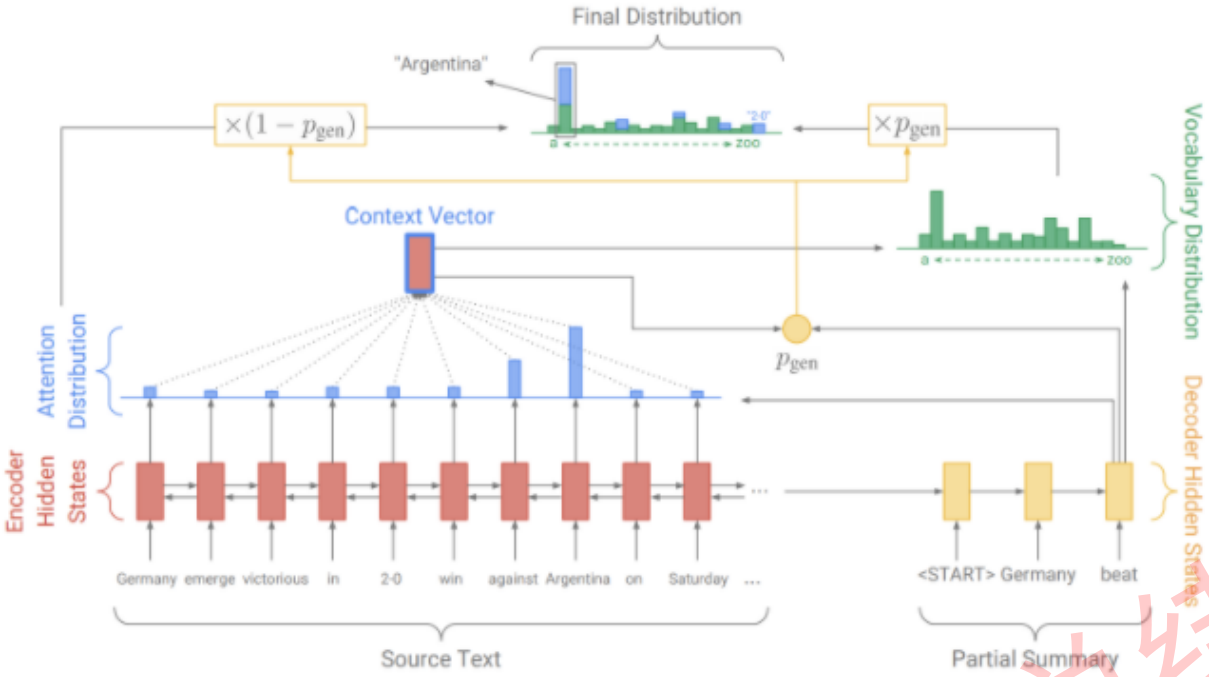
在训练阶段，时间步 t 时的损失为：

$$loss_t = -logP(w_t^*)$$

那么原输入序列的整体损失为：

$$loss = \frac{1}{T} \sum_{t=0}^{T} loss_t$$

5．Pointer-Generator-Network



如何权衡一个词应该是生成的还是复制的?

原文中引入了一个权重 p_{gen} 。

从Baseline seq2seq的模型结构中得到了s_t 和h_t^*，和解码器输入 x_t 一起来计算 p_{gen} ：

$$p_{gen} = \sigma(w_{h^*}^T h_t^* + w_s^T s_t + w_x^T x_t + b_{ptr})$$

这时，会扩充单词表形成一个更大的单词表--扩充单词表(将原文当中的单词也加入到其中)，该时间步的预测词概率为：

$$P(w) = p_{gen} P_{vocab}(w) + (1 - p_{gen}) \sum_{i:w_i=w} a_i^t$$

其中 a_i^t 表示的是原文档中的词。我们可以看到解码器一个词的输出概率有其是否拷贝是否生成的概率和决定。**当一个词不出现在常规的单词表上时 $P_{vocab}(w)$ 为0，当该词不出现在文档中 $\sum_{i:w_i=w} a_i^t$ 为0。**

6．Coverage mechanism

原文的特色是运用了Coverage Mechanism来解决重复生成文本的问题。

具体实现上，就是将先前时间步的注意力权重加到一起得到所谓的覆盖向量$c^t(coveragevector)$，用先前的注意力权重决策来影响当前注意力权重的决策，这样就避免在同一位置重复，从而避免重复生成文本。计算上，先计算coverage vector c^t：

$$c^t = \sum_{t'=0}^{t-1} a^{t'}$$

然后添加到注意力权重的计算过程中，c^t用来计算 e_i^t：

$$e_i^t = v^T tanh(W_h h_i + W_s s_t + w_c c_i^t + b_{attn})$$

同时，为coverage vector添加损失是必要的，coverage loss计算方式为：

$$covloss_t = \sum_i min(a_i^t, c_i^t)$$

这样coverage loss是一个有界的量 $covloss_t \leq \sum_i a_i^t = 1$ 。因此最终的LOSS为：

$$loss_t = -logP(w_t^*) + \lambda \sum_i min(a_i^t, c_i^t)$$

7．训练过程

优化器Adagrad，初始学习率0.15。hidden_dim为 256 ，词向量维度emb_dim为126，词汇表数目vocab_size为50K，batch_size设为16。模型有处理OOV能力，因此词汇表不用设置过大；在batch_size的选择上，显存小的同学建议设为8，否则会出现内存不够，难以训练。一开始我们未开启coverage模式，迭代500K后终止，可以看出模型已收敛。然后选择一个loss较好的模型即pointer-gen模型，在这个模型的基础之上，开启coverage模式之后继续训练40K后终止，得到收敛的模型即是pointer-gen+coverage了。

8．评估方法

摘要质量评价需要考虑一下三点：

- 决定原始文本最重要的、需要保留的部分
- 在自动文本摘要中识别出1中的部分
- 基于语法和连贯性（coherence）评价摘要的可读性（readability）

从这三点出发有人工评价和自动评价，本文只讨论一下更值得关注的自动评价。自动文档摘要评价方法分为两类：

- 内部评价方法（Intrinsic Methods）：提供参考摘要，以参考摘要为基准评价系统摘要的质量。系统摘要与参考摘要越吻合， 质量越高。
- 外部评价方法（Extrinsic Methods）：不提供参考摘要，利用文档摘要代替原文档执行某个文档相关的应用。

ROUGE是2004年由ISI的Chin-Yew Lin提出的一种自动摘要评价方法，现被广泛应用于DUC（Document Understanding Conference）的摘要评测任务中。ROUGE基于摘要中n元词(n-gram)的共现信息来评价摘要，是一种面向n元词召回率的评价方法。基本思想为由多个专家分别生成人工摘要，构成标准摘要集，将系统生成的自动摘要与人工生成的标准摘要相对比，通过统计二者之间重叠的基本单元(n元语法、词序列和词对)的数目，来评价摘要的质量。通过与专家人工摘要的对比，提高评价系统的稳定性和健壮性。该方法现已成为摘要评价技术的通用标注之一。

- https://www.cnblogs.com/zingp/p/11571593.html
  <https://www.cnblogs.com/zingp/p/11571593.html>
- https://blog.csdn.net/mr2zhang/article/details/90754134
  <https://blog.csdn.net/mr2zhang/article/details/90754134>