10.10 束搜索

上一节介绍了如何训练输入和输出均为不定长序列的编码器—解码器。本节我们介绍如何使用编码器—解码器来预测不定长的序列。

上一节里已经提到，在准备训练数据集时，我们通常会在样本的输入序列和输出序列后面分别附上一个特殊符号"<eos>"表示序列的终止。我们在接下来的讨论中也将沿用上一节的全部数学符号。为了便于讨论，假设解码器的输出是一段文本序列。设输出文本词典$\mathcal{Y}$（包含特殊符号"<eos>"）的大小为$|\mathcal{Y}|$，输出序列的最大长度为T'。所有可能的输出序列一共有$\mathcal{O}(|\mathcal{Y}|^{T'})$种。这些输出序列中所有特殊符号"<eos>"后面的子序列将被舍弃。

10.10.1 贪婪搜索

让我们先来看一个简单的解决方案：贪婪搜索（greedy search）。对于输出序列任一时间步t'，我们从$|\mathcal{Y}|$个词中搜索出条件概率最大的词

$$y_{t'} = \underset{y \in \mathcal{Y}}{\operatorname{argmax}} P(y \mid y_1, \ldots, y_{t'-1}, \boldsymbol{c})$$

作为输出。一旦搜索出"<eos>"符号，或者输出序列长度已经达到了最大长度T'，便完成输出。

我们在描述解码器时提到，基于输入序列生成输出序列的条件概率是$\prod_{t'=1}^{T'} P(y_{t'} \mid y_1, \ldots, y_{t'-1}, \boldsymbol{c})$。我们将该条件概率最大的输出序列称为最优输出序列。而贪婪搜索的主要问题是不能保证得到最优输出序列。

下面来看一个例子。假设输出词典里面有“A”“B”“C”和“<eos>”这4个词。图10.9中每个时间步下的4个数字分别代表了该时间步生成“A”“B”“C”和“<eos>”这4个词的条件概率。在每个时间步，贪婪搜索选取条件概率最大的词。因此，图10.9中将生成输出序列“A”“B”“C”“<eos>”。该输出序列的条件概率是$0.5 \times 0.4 \times 0.4 \times 0.6 = 0.048$。

时间步	1	2	3	4
A	0.5	0.1	0.2	0.0
B	0.2	0.4	0.2	0.2
C	0.2	0.3	0.4	0.2
<eos>	0.1	0.2	0.2	0.6

图10.9 在每个时间步，贪婪搜索选取条件概率最大的词

接下来，观察图10.10演示的例子。与图10.9中不同，图10.10在时间步2中选取了条件概率第二大的词“C”。由于时间步3所基于的时间步1和2的输出子序列由图10.9中的“A”“B”变为了图10.10中的“A”“C”，图10.10中时间步3生成各个词的条件概率发生了变化。我们选取条件概率最大的词“B”。此时时间步4所基于的前3个时间步的输出子序列为“A”“C”“B”，与图10.9中的“A”“B”“C”不同。因此，图10.10中时间步4生成各个

词的条件概率也与图10.9中的不同。我们发现，此时的输出序列“A”“C”“B”“<eos>”的条件概率是$0.5 \times 0.3 \times 0.6 \times 0.6 = 0.054$，大于贪婪搜索得到的输出序列的条件概率。因此，贪婪搜索得到的输出序列“A”“B”“C”“<eos>”并非最优输出序列。

时间步	1	2	3	4
A	0.5	0.1	0.1	0.1
B	0.2	0.4	0.6	0.2
C	0.2	0.3	0.2	0.1
<eos>	0.1	0.2	0.1	0.6

图10.10 在时间步2选取条件概率第二大的词“C”

10.10.2 穷举搜索

如果目标是得到最优输出序列，我们可以考虑穷举搜索（exhaustive search）：穷举所有可能的输出序列，输出条件概率最大的序列。

虽然穷举搜索可以得到最优输出序列，但它的计算开销$O(|Y|^{T'})$很容易过大。例如，当$|Y| = 10000$且$T' = 10$时，我们将评估$10000^{10} = 10^{40}$个序列：这几乎不可能完成。而贪婪搜索的计算开销是$O(|Y|T')$，通常显著小于穷举搜索的计算开销。例如，当$|Y| = 10000$且$T' = 10$时，我们只需评估$10000 \times 10 = 10^5$个序列。

10.10.3 束搜索

束搜索（beam search）是对贪婪搜索的一个改进算法。它有一个束宽（beam size）超参数。我们将它设为k。在时间步1时，选取当前时间步条件概率最大的k个词，分别组成k个候选输出序列的首词。在之后的每个时间步，基于上个时间步的k个候选输出序列，从$k|Y|$个可能的输出序列中选取条件概率最大的k个，作为该时间步的候选输出序列。最终，我们从各个时间步的候选输出序列中筛选出包含特殊符号“<eos>”的序列，并将它们中所有特殊符号“<eos>”后面的子序列舍弃，得到最终候选输出序列的集合。

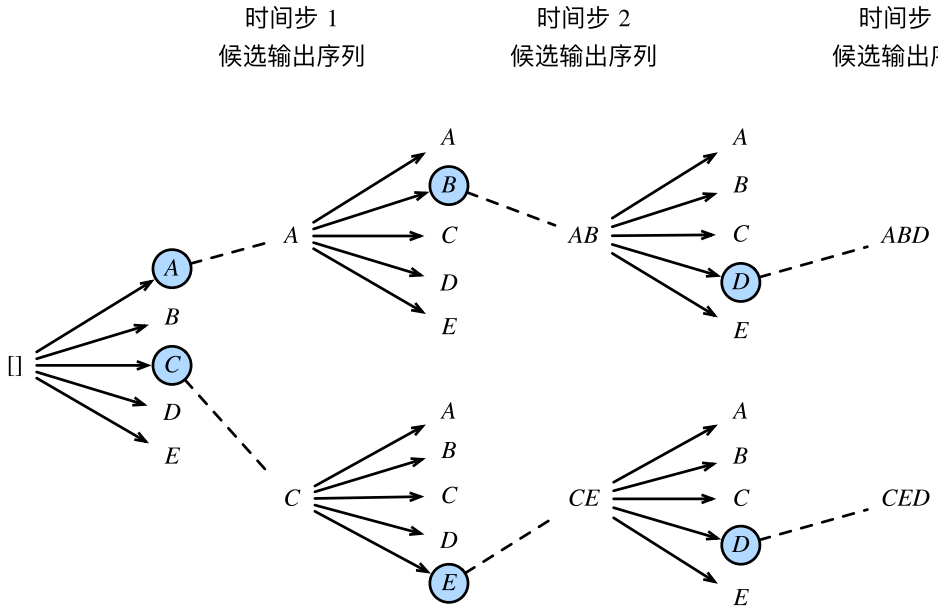

图10.11 束搜索的过程。束宽为2，输出序列最大长度为3。候选输出序列有A、C、AB、CE、ABD和CED

图10.11通过一个例子演示了束搜索的过程。假设输出序列的词典中只包含5个元素，即$\mathcal{Y} = \{A, B, C, D, E\}$，且其中一个为特殊符号“<eos>”。设束搜索的束宽等于2，输出序列最大长度为3。在输出序列的时间步1时，假设条件概率$P(y_1 \mid \boldsymbol{c})$最大的2个词为$A$和$C$。我们在时间步2时将对所有的$y_2 \in \mathcal{Y}$都分别计算$P(y_2 \mid A, \boldsymbol{c})$和$P(y_2 \mid C, \boldsymbol{c})$，并从计算出的10个条件概率中取最大的2个，假设为$P(B \mid A, \boldsymbol{c})$和$P(E \mid C, \boldsymbol{c})$。那么，我们在时间步3时将对所有的$y_3 \in \mathcal{Y}$都分别计算$P(y_3 \mid A, B, \boldsymbol{c})$和$P(y_3 \mid C, E, \boldsymbol{c})$，并从计算出的10个条件概率中取最大的2个，假设为$P(D \mid A, B, \boldsymbol{c})$和$P(D \mid C, E, \boldsymbol{c})$。如此一来，我们得到6个候选输出序列：（1）$A$；（2）$C$；（3）$A$、$B$；（4）$C$、$E$；（5）$A$、$B$、$D$和（6）$C$、$E$、$D$。接下来，我们将根据这6个序列得出最终候选输出序列的集合。

在最终候选输出序列的集合中，我们取以下分数最高的序列作为输出序列：

$$\frac{1}{L^\alpha} \log P(y_1, \ldots, y_L) = \frac{1}{L^\alpha} \sum_{t'=1}^{L} \log P(y_{t'} \mid y_1, \ldots, y_{t'-1}, \boldsymbol{c}),$$

其中L为最终候选序列长度，α一般可选为0.75。分母上的L^α是为了惩罚较长序列在以上分数中较多的对数相加项。分析可知，束搜索的计算开销为$\mathcal{O}(k \mid \mathcal{Y} \mid T)$。这介于贪婪搜索和穷举搜索的计算开销之间。此外，贪婪搜索可看作是束宽为1的束搜索。束搜索通过灵活的束宽k来权衡计算开销和搜索质量。

小结

- 预测不定长序列的方法包括贪婪搜索、穷举搜索和束搜索。
- 束搜索通过灵活的束宽来权衡计算开销和搜索质量。