第三章 统计语言模型

目录

3.1 概述

- **语言模型（Language Model, LM）：一种用于计算词序列（如：短语、句子、段落等）概率分布的模型。模型用于评估词序列的合理性。**

例

Sentence 1:美联储主席本·伯南克昨天告诉媒体7000亿美元的救助资金

Sentence 2:美主席联储本·伯南克告诉昨天媒体7000亿美元的资金救

Sentence 3:美主车席联储本·克告诉昨天公司媒体7000伯南亿美行元

思考: 哪个句子更像一个合理的句子？如何量化估计这句话的“合理程度”?

句子S由特定词序列$w_1, w_2, \cdots, w_n$组成，n其中表示句子的长度，语言模型可以表示为一个概率分布$P(S)$，$P(S)$表示句子出现的概率

$$P(w_1, w_2, \ldots, w_n) = P(w_1)P(w_2|w_1)P(w_3|w_1, w_2)\cdots P(w_n|w_1, w_2, \cdots w_{n-1})$$

$$P(S) = P(w_1, w_2, \ldots, w_n)$$

3.1 概述

- **统计语言模型（Statistic Language Model, SLM）：通过大规模文本数据的统计分析来描述词语、语句甚至整个文档的概率分布，用于评估句子或词序列是否符合自然语言的规范。**

1.**主要内容**: 利用大型计算机和大规模的文本语料库进行统计建模，分析词语之间的搭配和出现频率，从而推导出词语的概率分布。

2. **优势**：不依赖于人为定义的语法规则，从实际语料中学习和推断自然语言的规律，处理自然语言复杂性和动态性。

3.2 N-gram 模型

思考1: 语言模型的计算复杂度?

回顾 $P(S)$ 的计算过程：对于 $P(w_1, w_2, \dots, w_{i-1})$ 计算，需要统计 $w_1, w_2, \dots, w_{i-1}$ 共同出现的频率，而当 i 无限增大时，计算多个词共现频率的复杂度将呈指数级上升

思考2：如何简化$P(S)$的计算?

（一阶）马尔可夫假设：一阶马尔可夫语言模型认为任意一个词w_i 出现的概率仅与其前一个词 w_{i-1}相关。因此，文本序列 $P(S)$的概率可以简化为

$$P(S) \approx P(w_1) \prod_{i=2}^{L} P(w_i|w_{i-1})$$

基于 N－1 阶 马尔可夫链的统计语言模型：假设当前词的出现的概率与其前 N－1 个连续的词相关

N 元文法（N-gram）模型

3.2 N-gram 模型

- **N元文法模型 (N-gram Model)：给定一个词序列$S = (w_1, \dots, w_L)$，每个词w_i的出现概率仅依赖于它之前的 N−1 个词，即**

$$P(S) = P(w_1, w_2, \cdots, w_L) = \prod_{i=1}^{L} P(w_i | w_{i-(N-1)}, w_{i-(N-2)}, \cdots, w_{i-1}).$$

- 当 N = 1 时，称为一元文法（Unigram 或 Monogram），每个词出现的概率只与其自身的词频相关
- 当 N = 2 时，称为二元文法（Bigram），其基于 1 阶马尔可夫链构造
- 当 N = 3 时，称为三元文法（Trigram），其基于 2 阶马 尔可夫链构造

例：长度为 5 的序列$(w_1, w_2, \dots, w_5)$ 在一元文法、二元文法和三元文法下的概率分别为

$$P(w_1, w_2, \cdots, w_5) = P(w_1)P(w_2)P(w_3)P(w_4)P(w_5) \tag{3.5}$$

$$P(w_1, w_2, \cdots, w_5) = P(w_1)P(w_2|w_1)P(w_3|w_2)P(w_4|w_3)P(w_5|w_4) \tag{3.6}$$

$$P(w_1, w_2, \cdots, w_5) = P(w_1)P(w_2|w_1)P(w_3|w_1, w_2)P(w_4|w_2, w_3)P(w_5|w_3, w_4)$$

3.2 N-gram 模型

例 对于二元模型，每个词都与它左边的最近的一个词有关联，也就是对于P(A,B,C) = P(A)P(B|A)P(C|B)

比如语句：“猫，跳上，椅子”，P(A="猫"，B="跳上"，C="椅子") = P("猫")P(“跳上”)P("椅子")；其中各个词的数量数语料库M中统计的数量

	猫	跳上	椅子
	13	16	23

比如语句：“猫，跳上，椅子”，P(A="猫"，B="跳上"，C="椅子") = P("猫")P(“跳上”|“猫”)P("椅子"|“跳上”)；其中各个词的数量数语料库M中统计的数量

	猫	跳上	椅子
猫	0	9	1
跳上	0	3	15
椅子	0	0	0

依据这些图表一和图表二就可以求出P(A,B,C),也就是这个句子的合理的概率.

$$P(A,B,C) = P(A)P(B|A)P(C|B)$$

$$p(A) = 13/M$$

$$P(B|A) = 9/13$$

$$p(C|B) = 15/16$$

3.2 N-gram 模型

例 给定句子 “Marry sings a song”， 计算该句子的概率。

利用基于 Bigram 模型计算上述句子的概率为：

$$P(Mary\ sings\ a\ song) = p(Mary|<BOS>) \times p(sings|Mary) \times p(a|sings) \times p(song|a) \times p(<EOS>|song)$$

$P(Mary)$ 表示在给定 "BOS" （句子起始标记）的情况下， 下一个词是"$Mary$" “的概率； $P(sings|Mary)$ 表示在给定"$Mary$"的情况下， 下一个词是"$sings$"的概率，以此类推。

3.2 N-gram 模型

例 “随着人工智能技术的不断发展，通用人工智能和数字经济领域的 融合与交叉已成为科研和产业发展的重要趋势。通用人工智能与数字经济创新 团队依托金融智能与金融工程四川省重点实验室，面向国家“智改数转”重大 战略需求，聚焦行业大模型研究，致力于探索大模型技术在数字经济领域的应 用潜力，推动人工智能技术与数字经济产业的创新发展。”

根据上述文本，用极大似然估计计算后验概率:

1. ”人工智能” 出现了 4 次，“人工”作为第一个词出现了 4 次，因此计算P(智能|人工):

$$P(\text{智能}|\text{人工}) = \frac{4}{4} = 1$$

2. ”金融智能” 出现了 1 次，“金融”作为第一个词出现了 2 次，因此计算P(智能|金融)

$$P(\text{智能}|\text{金融}) = \frac{1}{2} = 0.5$$

3.3 平滑技术

- **数据稀疏问题：统计语言模型中，训练数据中某些词或短语可能从未出现，或其上下文信息不足， 导致模型在估计这些词序列概率时可能出现零概率问题**

例 在二元文法模型中，$S = P(w_i, w_i + 1) = p(w_i)P(w_i + 1|w_i)$，则$P(w_i + 1|w_i) = P(w_i, w_i + 1)/P(w_i)$; $P(w_i) = 0$ (基于w_i在文本中的频率的得到)

语料库

今天 上午 的 天气 很好
我 很 想 出去 运动
但 今天 上午 有 课程
训练营 明天 才 开始

假设使用Bigram模型，则

P(今天,没有,训练营)
*=P(今天)*P(没有|今天)*P(训练营|没有)*
*=2/19*0*0*
=0

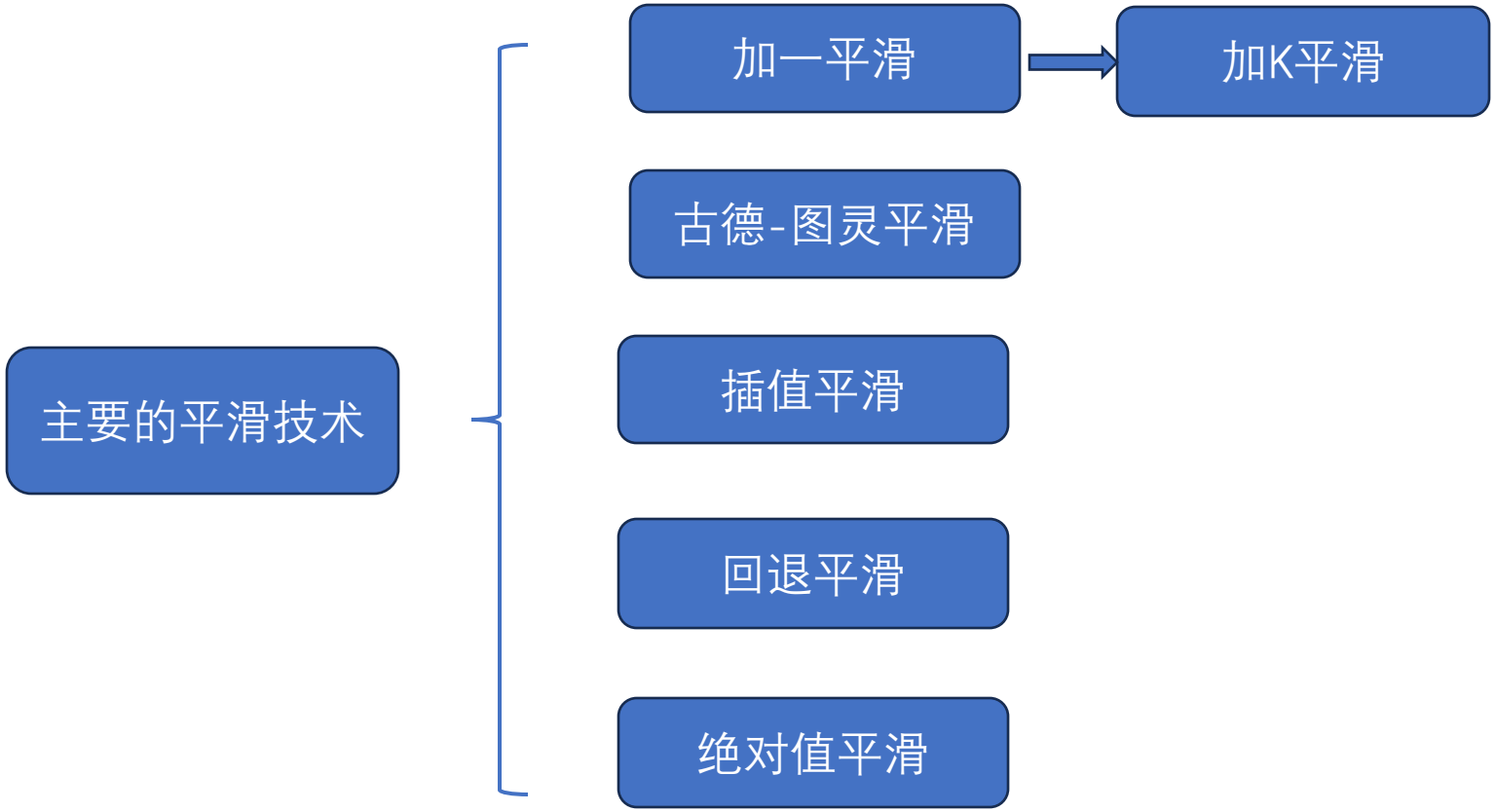
P(今天,训练营,没有)
*=P(今天)*P(训练营|今天)*P(没有|训练营)*
*=2/19*0*0*
=0

在上面的场景中，由于部分单词对出现的概率为0，导致最终两句话出现的概率均为0。但实际上，s1=“今天没有训练营”比s2=“今天训练营没有”更符合语法习惯，我们也更希望计算出来的P(s1)大于P(s2)。

为了解决上述问题，考虑引入平滑处理的技术，来修正计算过程中的概率值，避免某一项概率为0导致整个句子的概率为0。

3.3 平滑技术

- **平滑技术**：为那些在训练数据中未出现或出现次数极少的单词或短语提供一个**非零的概率估计**，从而使模型预测更为合理。



3.3 平滑技术：加一平滑

- **加一平滑（Add-one Smoothing）：通过将每个事件的观察次数增加一个小常数（通常为1）来减小高频事件的概率估计，然后将结果分配给低频事件，以平滑概率估计。**

基本思想: 每一种情况出现的次数加1。

例如，对于 Unigram，设 w_1, w_2, w_3 三个词，概率分别为：1/3, 0, 2/3，加1后情况？

2/6，1/6，3/6

例如，前面Bigram的例子：

语料库

今天 上午 的 天气 很好
我 很 想 出去 运动
但 今天 上午 有 课程
训练营 明天 才 开始

引入加一平滑

$$P_{\text{add1}}(w_i|w_{i-1}) = \frac{C(w_{i-1},w_i)+1}{C(w_{i-1})+|V|}$$

P_{add_1}(没有|今天)
={[P("今天没有")+1]/P(今天)+|V|}
= (0+1)/(2+19)
=1/21

3.3 平滑技术：加K平滑

- **加K平滑（Add-K Smoothing）：加一平滑的泛化形式。每个统计单元的频率计数增加了一个预定的常数K，而非单独增加1。数学表达如下：**

$$P(w_i|w_{i-1}) = \frac{\text{count}(w_{i-1}, w_i) + K}{\text{count}(w_{i-1}) + K \times |V|}.$$

加 K 平滑能够更灵活地调整平滑程度以适应不同的数据分布。具体来说，通过合适地选择 K 的值，可以在减少过度平滑与保持数据稳健性之间达到更好的平衡。

3.3 平滑技术：插值平滑

- **插值平滑（Interpolation Smoothing）：利用不同阶数的 N-gram 模型来估算概率。具体来说，插值平滑将各阶数模型的概率进行线性加权平均：**

$$P(w_i \mid w_{i-(N-1)}, \ldots, w_{i-1}) = \sum_{j=1}^{N} \lambda_j P(w_i \mid w_{i-j+1}, \ldots, w_{i-1})$$

其中，$\lambda_1, \lambda_2, \ldots, \lambda_N$ 是不同阶数的 N 元文法模型对应的权重系数。这种方法充分利用不同阶数的马尔可夫模型，但依赖于权重系数的选择。

3.3 平滑技术：绝对值平滑

- **绝对值平滑（Absolute Discounting）：直接从每个 N 元 文法事件的观察频率中减去一个固定的值 d，之后将剩余的概率质量分配给未见或低频事件。其数学表达式如下：**

$$P(w_i|w_{i-1}) = \frac{\max\left[\text{count}(w_{i-1}, w_i) - d, 0\right]}{\text{count}(w_{i-1})}$$

其中，$\text{count}(w_{i-1}, w_i)$ 是训练数据中 N 元文法模型中 (w_{i-1}, w_i)的出现次数，$\text{count}(w_{i-1})$ 是训练数据中以 w_{i-1} 开头的所有 N 元文法模型的总数。这种方法的一个优点是在处理稀疏数据时更有效，但模型效果依赖于减值参数 d 的选择。

3.4 讨论

- **讨论3.1：**

你认为语言模型在未来会有怎样的发展趋势？请提出至少亮点预测或观点支持你的观点。

- **讨论3.2：**

了解回退平滑技术。分析回退平滑和绝对值平滑分别适用于什么样的数据分布和模型需求？在面对大量未见事件或极端稀疏数据时，这两种平滑技术有哪些优势和局限性？