第5章 语言模型

北京市海淀区中关村东路95号
邮编：100190

电话：+86-10-6255 4263
邮件：cqzong@nlpr.ia.ac.cn

5.1 基本概念

5.1 基本概念

大规模语料库的出现为自然语言统计处理方法的实现提供了可能，统计方法的成功使用推动了语料库语言学的发展。

基于大规模语料库和统计方法，可以

- 发现语言使用的普遍规律
- 进行机器学习、自动获取语言知识
- 对未知语言现象进行推测

5.1 基本概念

如何计算一段文字(句子)的概率？

阳春三月春意盎然，少先队员脸上荡漾着喜悦的笑容，鲜艳的红领巾在他们的胸前迎风飘扬。

- 以一段文字(句子)为单位统计相对频率？
- 根据句子构成单位的概率计算联合概率？

$$p(w_1)\times p(w_2)\times\ldots\times p(w_n)$$

5.1 基本概念

语句 $s = w_1\, w_2 \ldots w_m$ 的先验概率：

$$P(s) = P(w_1) \times P(w_2|w_1) \times P(w_3|w_1 w_2) \times \ldots \times P(w_m|w_1 \ldots w_{m-1})$$

$$= \prod_{i=1}^{m} P(w_i \mid w_1 \cdots w_{i-1}) \qquad \ldots (5\text{-}1)$$

当 $i=1$ 时，$P(w_1|w_0) = P(w_1)$。

语言模型

5.1 基本概念

说明：

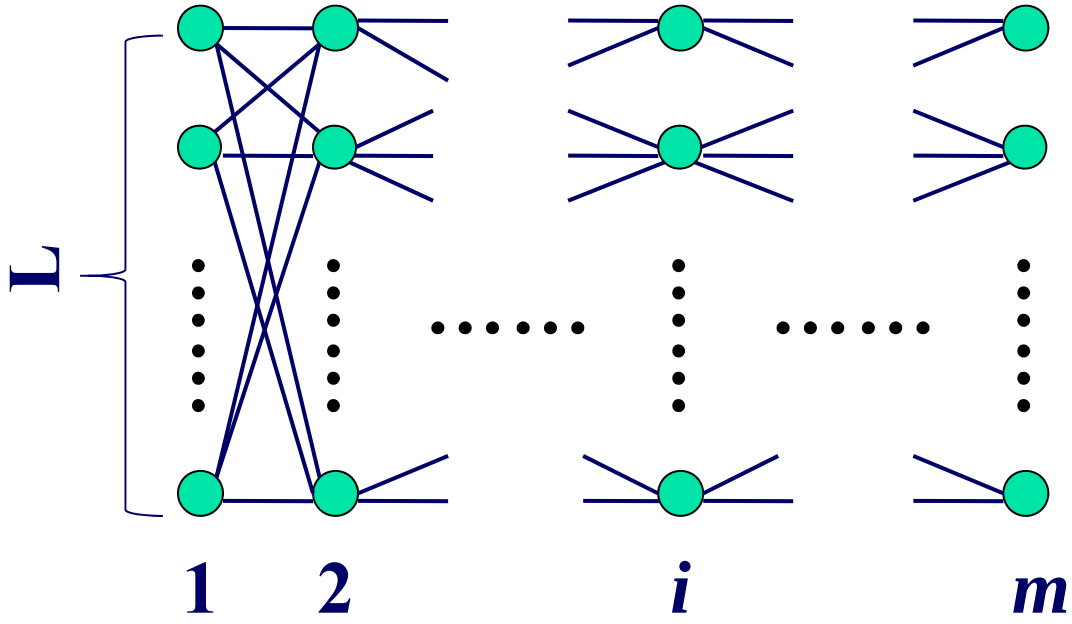
(1) w_i 可以是字、词、短语或词类等等，称为统计基元。通常以“词”代之。

(2) w_i 的概率由 $w_1, \ldots, w_{i-1}$ 决定，由特定的一组$w_1, \ldots, w_{i-1}$ 构成的一个序列，称为 w_i 的历史（history）。

5.1 基本概念

问题：随着历史基元数量的增加，不同的“历史”(路径)按指数级增长。对于第 i ($i>1$) 个统计基元，历史基元的个数为 i-1，如果共有 L 个不同的基元，如词汇表，理论上每一个单词都有可能出现在1到 i-1的每一个位置上，那么，i 基元就有 L^{i-1} 种不同的历史情况。我们必须考虑在所有的 L^{i-1} 种不同历史情况下产生第 i 个基元的概率。那么，模型中有L^m 个自由参数 $P(w_m|w_1 \ldots w_{m-1})$。

5.1 基本概念



如果 L=5000, m = 3, 自由参数的数目为 1250 亿！

5.1 基本概念

❒ 问题解决方法

设法减少历史基元的个数，将 $w_1\ w_2 \ldots w_{i-1}$ 映射到等价类 $S(w_1\ w_2 \ldots w_{i-1})$，使等价类的数目远远小于原来不同历史基元的数目。则有：

$$P(w_i \mid w_1, \cdots, w_{i-1}) = P(w_i \mid S(w_1, \cdots, w_{i-1}))$$

… (5-2)

5.1 基本概念

❒ 如何划分等价类

将两个历史映射到同一个等价类，当且仅当这两个历史中的最近 ***n*-1** 个基元相同，即：

$$H_1: w_1\, w_2 \cdots\cdots w_{i-n+2}\, w_{i-n+3} \cdots w_{i-1}\, w_i \cdots\cdots$$

$$H_2: \quad v_1\, v_2 \cdots\cdots v_{k-n+2}\, v_{k-n+3} \cdots v_{k-1}\, v_k \cdots\cdots$$

n-1

$$S(w_1, w_2, \cdots, w_i) = S(v_1, v_2 \cdots, v_k)$$

$$iff \quad H_1:(w_{i-n+2}, \cdots; w_i) = H_2:(v_{k-n+2}, \cdots; v_k) \qquad \ldots (5\text{-}3)$$

5.1 基本概念

这种情况下的语言模型称为 ***n* 元文法(*n*-gram)**。

通常地，

- 当 $n=1$ 时，即出现在第 i 位上的基元 w_i 独立于历史。
  一元文法也被写为 uni-gram 或 monogram;
- 当 $n=2$ 时, **2-gram** (bi-gram) 被称为1阶马尔柯夫链;
- 当 $n=3$ 时, **3-gram**(tri-gram)被称为2阶马尔柯夫链，
  依次类推。

5.1 基本概念

为了保证条件概率在 $i=1$ 时有意义，同时为了保证句子内所有字符串的概率和为 1，即 $\sum_s p(s)=1$，可以在句子首尾两端增加两个标志：<BOS> $w_1\ w_2 \ldots w_m$ <EOS>。不失一般性，对于$n>2$的 n-gram，$P(s)$ 可以分解为：

$$P(s)=\prod_{i=1}^{m+1} P(w_i \mid w_{i-n+1}^{i-1}) \qquad \ldots (5\text{-}4)$$

其中，w_i^j 表示词序列 $w_i \ldots w_j$，w_{i-n+1} 从 w_0 开始，w_0 为 <BOS>，w_{m+1} 为 <EOS>。

5.1 基本概念

- 举例：

给定句子：John read a book

增加标记：<BOS> John read a book <EOS>

Unigram：<BOS>, John, read, a, book, <EOS>

Bigram：(<BOS>John), (John read), (read a), (a book), (book <EOS>)

Trigram：(<BOS>John read), (John read a), (read a book), (a book <EOS>)

5.1 基本概念

<BOS> John read a book <EOS>

基于2元文法的概率为：

$$P(\text{John read a book}) = P(\text{John}|\text{<BOS>}) \times$$
$$P(\text{read}|\text{John}) \times P(\text{a}|\text{read}) \times$$
$$P(\text{book}|\text{a}) \times P(\text{<EOS>}|\text{book})$$

5.1 基本概念

❑ 应用－1：音字转换问题

给定拼音串：ta shi yan jiu sheng wu de

可能的汉字串：踏实研究生物的

他实验救生物的

他使烟酒生物的

他是研究生物的

… …

5.1 基本概念

$$\begin{aligned}\hat{C}String &= \arg\max_{CString} P(CString \mid Pinyin) \\ &= \arg\max_{CString} \frac{P(Pinyin \mid CString)P(CString)}{P(Pingyin)} \\ &= \arg\max_{CString} P(Pinyin \mid CString)P(CString) \\ &= \arg\max_{CString} P(CString)\end{aligned}$$

5.1 基本概念

CString = {踏实研究生物的，他实验救生物的，他是研究生物的，他使烟酒生雾的，… …}

如果使用 **2-gram**：

P(***CString***$_1$) =P(踏实|<BOS>) × P(研究|踏实) ×
P(生物|研究) × P(的|生物) × P(<EOS>|的)

P(***CString***$_2$) =P(他|<BOS>) × P(实验|他) × P(救|实验) ×
P(生物|救) × P(的|生物) × P(<EOS>|的)

……

5.1 基本概念

如果汉字的总数为：N

- 一元语法：1）样本空间为 N
  2）只选择使用频率最高的汉字
- 2元语法：1）样本空间为 N^2
  2）效果比一元语法明显提高
- 估计对汉字而言四元语法效果会好一些
- 智能狂拼、微软拼音输入法基于 ***n*-gram.**

5.1 基本概念

❑ 应用—2：汉语分词问题

给定汉字串：他是研究生物的。

可能的汉字串：

1）他|是|研究生|物|的

2）他|是|研究|生物|的

5.1 基本概念

$$\widehat{Seg} = \arg\max_{Seg} P(Seg \mid Text)$$

$$= \arg\max_{Seg} \frac{P(Text \mid Seg)P(Seg)}{P(Text)}$$

$$= \arg\max_{Seg} P(Text \mid Seg)P(Seg)$$

$$= \arg\max_{Seg} P(Seg)$$

5.1 基本概念

如果采用2元文法：

$$P(Seg1) = P(他|<BOS>) \times P(是|他) \times P(研究生|是) \times P(物|研究生) \times P(的|物) \times P(的|<EOS>)$$

$$P(Seg2) = P(他|<BOS>) \times P(是|他) \times P(研究|是) \times P(生物|研究) \times P(的|生物) \times P(的|<EOS>)$$

问题：如何获得 *n* 元语法模型？

5.2 参数估计

5.2 参数估计

❑两个重要概念：

- 训练语料(*training data*)：用于建立模型，确定模型参数的已知语料。
- 最大似然估计(*maximum likelihood Evaluation*, MLE)：用相对频率计算概率的方法。

5.2 参数估计

对于 ***n*-gram**，参数 $P(w_i \mid w_{i-n+1}^{i-1})$ 可由最大似然估计求得：

$$P(w_i \mid w_{i-n+1}^{i-1}) = f(w_i \mid w_{i-n+1}^{i-1}) = \frac{c(w_{i-n+1}^{i})}{\sum_{w_i} c(w_{i-n+1}^{i})} \quad \ldots(5\text{-}5)$$

其中，$\sum_{w_i} c(w_{i-n+1}^{i})$ 是历史串 w_{i-n+1}^{i-1} 在给定语料中出现的次数，即 $c(w_{i-n+1}^{i-1})$，不管 w_i 是什么。

$f(w_i \mid w_{i-n+1}^{i-1})$ 是在给定 w_{i-n+1}^{i-1} 的条件下 w_i 出现的相对频度，分子为 w_{i-n+1}^{i-1} 与 w_i 同现的次数。

5.2 参数估计

例如，给定训练语料：

“*John read Moby Dick*”，

“*Mary read a different book*”，

“*She read a book by Cher*”

根据 2 元文法求句子的概率？

5.2 参数估计

$$P(John|<BOS>)=\frac{c(<BOS>\ John)}{\sum_w c(<BOS>\ w)}=\frac{1}{3} \qquad P(a\,|\,read)=\frac{c(read\quad a)}{\sum_w c(read\quad w)}=\frac{2}{3}$$

$$P(read\,|\,John)=\frac{c(John\quad read)}{\sum_w c(John\quad w)}=\frac{1}{1} \qquad P(book\,|\,a)=\frac{c(a\quad book)}{\sum_w c(a\quad w)}=\frac{1}{2}$$

$$P(<EOS>|book)=\frac{c(book\quad <EOS>)}{\sum_w c(book\quad w)}=\frac{1}{2}$$

$$P(John\ read\ a\ book)=\frac{1}{3}\times 1\times\frac{2}{3}\times\frac{1}{2}\times\frac{1}{2}\approx 0.06$$

<BOS>John read Moby Dick<EOS>

<BOS>Mary read a different book<EOS>

<BOS>She read a book by Cher<EOS>

5.2 参数估计

P(*Cher read a book*) = ?

$$=P(Cher|<BOS>)\times P(read|Cher)\times P(a|read)\times$$
$$P(book|a)\times P(<EOS>|book)$$

$$P(Cher\ |<BOS>)=\frac{c(<BOS>\ Cher)}{\sum_w c(<BOS>\ w)}=\frac{0}{3}$$

$$P(read\,|\,Cher)=\frac{c(Cher\quad read)}{\sum_w c(Cher\quad w)}=\frac{0}{1}$$

于是， *P*(*Cher read a book*) = 0

<BOS>John read Moby Dick<EOS>

<BOS>Mary read a different book<EOS>

<BOS>She read a book by Cher<EOS>

5.2 参数估计

问题：

数据匮乏(稀疏) (*Sparse Data*) 引起零概率问题，如何解决？

数据平滑(data smoothing)

5.3 数据平滑

5.3 数据平滑

- 数据平滑的基本思想：

  调整最大似然估计的概率值，使零概率增值，使非零概率下调，“劫富济贫”，消除零概率，改进模型的整体正确率。

- 基本目标：测试样本的语言模型困惑度越小越好。

- 基本约束：$\sum_{w_i} P(w_i \mid w_1, w_2, \cdots, w_{i-1}) = 1$

5.3 数据平滑

➢ 回顾－困惑度的定义：

对于一个平滑的 ***n*-gram**，其概率为 $P(w_i \mid w_{i-n+1}^{i-1})$，

可以计算句子的概率：

$$P(s) = \prod_{i=1}^{m+1} P(w_i \mid w_{i-n+1}^{i-1})$$

假定测试语料 T 由 l_T 个句子构成 $(t_1, \ldots, t_{lT})$，则整个测试集的概率为：

$$P(T) = \prod_{i=1}^{l_T} P(t_i)$$

5.3 数据平滑

模型 $P(w_i \mid w_{i-n+1}^{i-1})$ 对于测试语料的交叉熵：

$$H_p(T) = -\frac{1}{W_T}\log_2 P(T)$$

其中，W_T 是测试文本 T 的词数。

模型 P 的困惑度 $PP_P(T)$ 定义为：

$$PP_P(T) = 2^{H_P(T)}$$

n-gram 对于英语文本的困惑度范围一般为 50～1000，对应于交叉熵范围为 6－10 bits/word。

5.3 数据平滑

❑加1法 (Additive smoothing)

基本思想：每一种情况出现的次数加1。

例如，对于 ***uni*-gram**，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

5.3 数据平滑

对于**2-gram** 有：

$$P(w_i \mid w_{i-1}) = \frac{1+c(w_{i-1}w_i)}{\sum_{w_i}[1+c(w_{i-1}w_i)]}$$

$$= \frac{1+c(w_{i-1}w_i)}{|V|+\sum_{w_i} c(w_{i-1}w_i)}$$

其中，V 为被考虑语料的词汇量（全部可能的基元数）。

5.3 数据平滑

在前面 3 个句子的例子中,

$$P(Cher\ read\ a\ book) = P(Cher|<BOS>)\times$$
$$P(read|Cher)\times P(a|read)\times P(book|a)\times$$
$$P(<EOS>|book)$$

<BOS>John read Moby Dick<EOS>
<BOS>Mary read a different book<EOS>
<BOS>She read a book by Cher<EOS>

原来:

P(*Cher*|<*BOS*>) = 0/3

P(*read*|*Cher*) = 0/1

P(*a*|*read*) = 2/3

P(*book*|*a*) = 1/2

P(<*EOS*>|*book*)= 1/2

5.3 数据平滑

词汇量：$|V|=11$

<BOS>John read Moby Dick<EOS>

<BOS>Mary read a different book<EOS>

<BOS>She read a book by Cher<EOS>

平滑以后：

P(*Cher*|<*BOS*>) = (0+1)/(11+3) = 1/14

P(*read*|*Cher*) = (0+1)/(11+1) = 1/12

P(*a*|*read*) = (1+2)/(11+3) = 3/14

P(*book*|*a*) = (1+1)/(11+2) = 2/13

P(<*EOS*>|*book*)= (1+1)/(11+2) = 2/13

$$P(\textit{Cher read a book}) = \frac{1}{14}\times\frac{1}{12}\times\frac{3}{14}\times\frac{2}{13}\times\frac{2}{13}\approx 0.00003$$

5.3 数据平滑

同理，对于句子*John read a book* 数据平滑后：

$P(John|<BOS>) = 2/14$, $P(read|John) = 2/12$,

$P(a|read) = 3/14$, $P(book|a) = 2/13$, $P(<EOS>|book) = 2/13$

于是，$P(John\ read\ a\ book) = P(John|<BOS>) \times P(read|John) \times P(a|read) \times P(book|a) \times P(<EOS>|book)$

$$= \frac{2}{14} \times \frac{2}{12} \times \frac{3}{14} \times \frac{2}{13} \times \frac{2}{13} \approx 0.0001$$

<BOS>John read Moby Dick<EOS>

<BOS>Mary read a different book<EOS>

<BOS>She read a book by Cher<EOS>

5.3 数据平滑

- **减值法/折扣法 (Discounting)**

基本思想：修改训练样本中事件的实际计数，使样本中(实际出现的)不同事件的概率之和小于1，剩余的概率量分配给未见概率。

5.3 数据平滑

(1) Good-Turing 估计

I. J. Good 1953 年引用 **Turing** 的方法来估计概率分布。

假设 N 是原来训练样本数据的大小， n_r 是在样本中正好出现 r 次的事件的数目(在这里，事件为 n-gram $w_1, w_2, \ldots, w_n$)，即：出现 1 次的 n_1个，出现 2 次的 n_2个, ……

5.3 数据平滑

那么， $N=\sum_{r=1}^{\infty} n_r r$ … (5-6)

由于，$N=\sum_{r=0}^{\infty} n_r r^* = \sum_{r=0}^{\infty}(r+1)n_{r+1}$ 所以，$r^* = (r+1)\frac{n_{r+1}}{n_r}$

那么，**Good-Turing** 估计在样本中出现 ***r*** 次的事件的概率为：

$$P_r = \frac{r^*}{N} \quad \text{… (5-7)}$$

5.3 数据平滑

实际应用中，一般直接用 n_{r+1} 代替 $E(n_{r+1})$，n_r 代替 $E(n_r)$。这样，原训练样本中所有事件的概率之和为：

$$\sum_{r>0} n_r \times P_r = 1 - \frac{n_1}{N} < 1 \qquad \ldots (5\text{-}8)$$

因此，有 $\frac{n_1}{N}$ 的剩余的概率量就可以均分给所有的未见事件 $(r = 0)$。

Good-Turing 估计适用于大词汇集产生的符合多项式分布的大量的观察数据。

有关证明和推导，请参阅：*A. Nadas. on Turing's Formula for Word Probabilities. In IEEE Trans. On ASSP-33, Dec. 1985. Pages 1414-1416.*

5.3 数据平滑

举例说明：假设有如下英语文本，估计 2-gram概率：

<BOS>John read Moby Dick<EOS>
<BOS>Mary read a different book<EOS>
<BOS>She read a book by Cher<EOS>
……

从文本中统计出不同 2-gram 出现的次数：

<BOS> John	**15**
<BOS> Mary	**10**
……	
read Moby	**5**
……	

5.3 数据平滑

假设要估计以 read 开始的 2-gram 概率，列出以read开始的所有 2-gram，并转化为频率信息：

r	n_r	r^*
1	2053	0.446
2	458	1.25
3	191	2.24
4	107	3.22
5	69	4.17
6	48	5.25
7	36	保持原来的计数

$$r^* = (r+1)\frac{n_{r+1}}{n_r}$$

$n_{r+1} = 0$

5.3 数据平滑

得到 r^* 后，就可以应用公式(5-7) 计算概率：

$$P_r = \frac{r^*}{N} \quad \ldots (5\text{-}7)$$

其中，N 为以read开始bigram的样本空间，即read出现的次数。那么，以read作为历史，没有出现过的 2-gram的概率总和为：

$$P_0 = \frac{n_1}{N}$$

以 read作为历史，没有出现过的 2-gram的个数等于：

$$n_0 = |V_T| - \sum_{r>0} n_r$$

其中，$|V_T|$ 为语料的词汇量。

5.3 数据平滑

那么，没有出现过的那些 2-gram的概率为：$\frac{P_0}{n_0}$。

注意：$\sum_{r=0}^{7} P_r \neq 1$

因此，需要归一化处理：$\hat{P}_r = \frac{P_r}{\sum_r P_r}$

5.3 数据平滑

(2) Back-off （后备/后退）方法

S. M. Katz 于 1987 年提出，所以又称 Katz 后退法。

基本思想：当某一事件在样本中出现的频率大于 ***K*** **(通常取 *K* 为0 或1)时，运用最大似然估计减值来估计其概率，否则，使用低阶的，即 (*n*-1)gram 的概率替代 *n*-gram 概率，而这种替代需受归一化因子α的作用。**

5.3 数据平滑

Back-off 方法的另一种理解:

对每个计数 $r > 0$ 的减值，把因减值而节省下来的剩余概率根据低阶的 **(*n*-1)gram** 分配给未见事件。

5.3 数据平滑

以2元文法模型为例，说明Katz平滑方法 。

对于一个出现次数为 $r = c(w_{i-1}^{i})$ 的 2元语法 w_{i-1}^{i}

使用如下公式计算修正的计数：

$$c_{\text{katz}}(w_{i-1}^{i}) = \begin{cases} d_r r & if \quad r > 0 \\ \alpha(w_{i-1}) p_{ML}(w_i) & if \quad r = 0 \end{cases}$$

也就是说，所有具有非零计数r 的 2元语法都根据折扣率d_r 被减值了，折扣率d_r 近似地等于 r^*/r，减值由Good-Turing估计方法预测。

5.3 数据平滑

从非零计数中减去的计数量，根据低一阶的分布，即一元语法模型被分配给了计数为零的 2 元语法。那么，需要选择 $\alpha(w_{i-1})$值，使分布中总的计数 $\sum_{w_i} c_{\text{katz}}(w_{i-1}^{i})$保持不变，即

$$\sum_{w_i} c_{\text{katz}}(w_{i-1}^{i}) = \sum_{w_i} c(w_{i-1}^{i})$$

$$c_{\text{katz}}(w_{i-1}^{i}) = \begin{cases} d_r r & if \quad r > 0 \\ \alpha(w_{i-1}) p_{ML}(w_i) & if \quad r = 0 \end{cases}$$

5.3 数据平滑

$\alpha(w_{i-1})$ 的适当值取为：

$$\alpha(w_{i-1}) = \frac{1 - \sum_{w_i:\ c(w_{i-1}^i)>0} p_{\text{katz}}(w_i \mid w_{i-1})}{\sum_{w_i:\ c(w_{i-1}^i)=0} p_{\text{ML}}(w_i)}$$

$$= \frac{1 - \sum_{w_i:c(w_{i-1}^i)>0} p_{\text{katz}}(w_i \mid w_{i-1})}{1 - \sum_{w_i:c(w_{i-1}^i)>0} p_{\text{ML}}(w_i)}$$

5.3 数据平滑

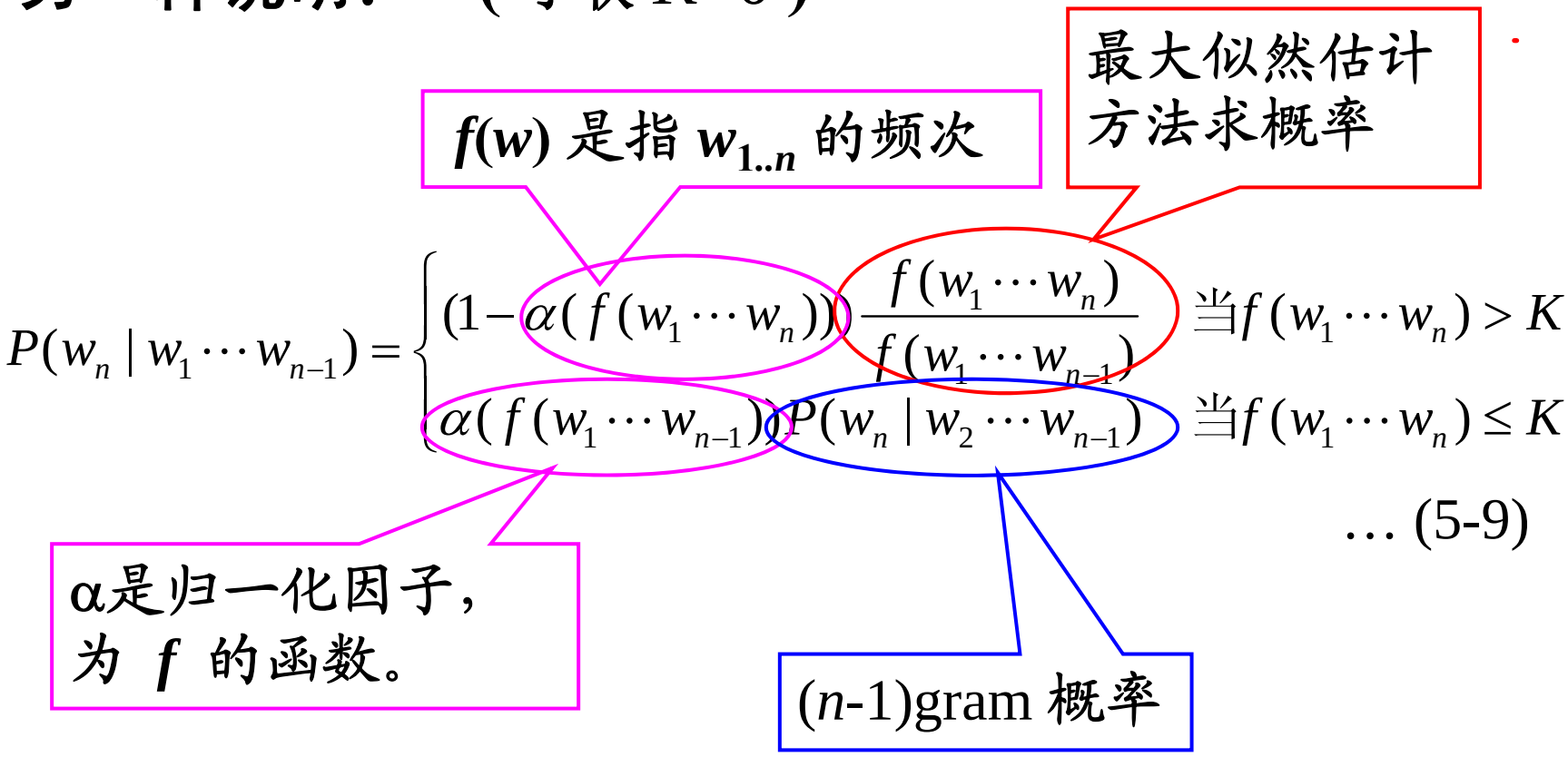
要根据修正的计数计算概率 $p_{\text{katz}}(w_i \mid w_{i-1})$

只需要归一化：

$$p_{\text{katz}}(w_i \mid w_{i-1}) = \frac{c_{\text{katz}}(w_{i-1}^i)}{\sum_{w_i} c_{\text{katz}}(w_{i-1}^i)}$$

5.3 数据平滑

另一种说明： (可取 K=0)



5.3 数据平滑

(3)绝对减值法(Absolute discounting)

Hermann Ney 和 U. Essen 提出。

基本思想：从每个计数 r 中减去同样的量，剩余的概率量由未见事件均分。

设 K 为所有可能事件的数目（当事件为 n-gram 时，如果统计基元为词，且词汇集的大小为 L，则 $K=L^n$ ）。

5.3 数据平滑

那么，样本出现了 r 次的事件的概率可以由如下公式估计：

$$P_r = \begin{cases} \dfrac{r-b}{N} & \text{当 } r > 0 \\ \dfrac{b(K-n_0)}{Nn_0} & \text{当 } r = 0 \end{cases} \quad \ldots (5\text{-}10)$$

其中，n_0 为样本中未出现的事件的数目。b 为减去的常量。

5.3 数据平滑

$b(K - n_0)/N$ 是由于减值而产生的剩余概率量。

b 为自由参数，可以通过留存数据(heldout data)法求得, 利用留一法(leave one out) 可以求得***b***的上限为：

$$b \le \frac{n_1}{n_1 + 2n_2} < 1 \qquad \ldots (5\text{-}11)$$

实际运用中，常用上限代替优化的 ***b***。

H. Ney and U. Essen. Estimating Small Probabilities by Leaving-one-Out. In Proc. Eurospeech 1993. Pages 2239-2242.

5.3 数据平滑

(4) 线性减值法

基本思想：从每个计数 r 中减去与该计数成正比的量(减值函数为线性的)，剩余概率量α被n_0个未见事件均分。

$$P_r = \begin{cases} \dfrac{(1-\alpha)r}{N} & \text{当 } r > 0 \\ \dfrac{\alpha}{n_0} & \text{当 } r = 0 \end{cases} \quad \ldots (5\text{-}12)$$

自由参数 α 的优化值为：$\dfrac{n_1}{N}$

绝对减值法产生的*n*-gram 通常优于线性减值法。

5.3 数据平滑

❑ 删除插值法 (Deleted interpolation)

基本思想：用低阶语法估计高阶语法，即当 3-gram 的值不能从训练数据中准确估计时，用 2-gram 来替代，同样，当 2-gram 的值不能从训练语料中准确估计时，可以用 1-gram 的值来代替。插值公式：

$$P(w_3 \mid w_1 w_2) = \lambda_3 P'(w_3 \mid w_1 w_2) + \lambda_2 P'(w_3 \mid w_2) + \lambda_1 P'(w_3)$$

其中，$\lambda_1 + \lambda_2 + \lambda_3 = 1$ … (5-13)

5.3 数据平滑

- λ_1，λ_2，λ_3 的确定：

将训练语料分为两部分，即从原始语料中删除一部分作为留存数据（heldout data）。

第一部分用于估计 $P'(w_3 \mid w_1 w_2)$，$P'(w_3 \mid w_2)$ 和 $P'(w_3)$ 。

第二部分用于计算 $\lambda_1, \lambda_2, \lambda_3$：使语言模型对留存数据的困惑度（Perplexity）最小。

5.3 数据平滑

- 各种平滑方法的详细介绍和比较请参阅：

  Chen, Stanley F. and Joshua Goodman. 1998. An Empirical Study of Smoothing Techniques for Language Model. Available from the website: *http://www-2.cs.cmu.edu/~sfc/html/publications.html*

- SRI 语言模型工具：

  http://www.speech.sri.com/projects/srilm/

- CMU-Cambridge 语言模型工具：

  http://mi.eng.cam.ac.uk/~prc14/toolkit.html

5.4 语言模型的自适应

5.4 语言模型的自适应

问题：

① 在训练语言模型时所采用的语料往往来自多种不同的领域，这些综合性语料难以反映不同领域之间在语言使用规律上的差异，而语言模型恰恰对于训练文本的类型、主题和风格等都十分敏感；

② n 元语言模型的独立性假设的前提是一个文本中的当前词出现的概率只与它前面相邻的 n-1 个词相关，但这种假设在很多情况下是明显不成立的。

5.4 语言模型的自适应

自适应方法:

- 基于缓存的语言模型 (cache-based LM)
- 基于混合方法的语言模型
- 基于最大熵的语言模型

5.4 语言模型的自适应

◆基于缓存的语言模型（Cache-based LM）

该方法针对的问题是：在文本中刚刚出现过的一些词在后边的句子中再次出现的可能性往往较大，比标准的 ***n*-gram** 模型预测的概率要大。针对这种现象，**cache-based**自适应方法的基本思路是：语言模型通过 ***n*-gram** 的线性插值求得：

$$\hat{P}(w_i \mid w_1^{i-1}) = \lambda \hat{P}_{Cache}(w_i \mid w_1^{i-1}) + (1-\lambda)\hat{P}_{n-gram}(w_i \mid w_{i-n+1}^{i-1}) \quad \cdots (5\text{-}14)$$

插值系数λ 可以通过**EM**算法求得。

5.4 语言模型的自适应

通常的处理方法是：在缓存中保留前面的 ***K***个单词，每个词的概率（缓存概率）用其在缓存中出现的相对频率计算得出：

$$\hat{P}_{Cache}(w_i \mid w_1^{i-1}) = \frac{1}{K} \sum_{j=i-K}^{i-1} I_{\{w_j = w_i\}} \qquad \ldots (5\text{-}15)$$

其中，I_ε为指示器函数(indicator function)，如果 ε 表示的情况出现，则 $I_\varepsilon = 1$，否则，$I_\varepsilon = 0$。

5.4 语言模型的自适应

这种方法的缺陷是，缓存中一个词的重要性独立于该词与当前词的距离。**P. R. Clarkson**等人**(1997)** 的研究表明，缓存中每个词对当前词的影响随着与该词距离的增大呈指数级衰减，因此，将**(5-15)** 式写成：

$$\hat{P}_{Cache}(w_i \mid w_1^{i-1}) = \beta \sum_{j=1}^{i-1} I_{\{w_i = w_j\}} e^{-\alpha(i-j)} \qquad \ldots(5\text{-}16)$$

其中，α 为衰减率，β为归一化常数，以使得

$\sum_{w_i \in V} P_{Cache}(w_i \mid w_1^{i-1}) = 1$，**V**为词汇表。

5.4 语言模型的自适应

◆基于混合方法的语言模型

该方法针对的问题是：由于大规模训练语料本身是异源的(heterogenous)，来自不同领域的语料无论在主题(topic)方面，还是在风格(style)方面，或者两者都有一定的差异，而测试语料一般是同源的(homogeneous)，因此，为了获得最佳性能，语言模型必须适应各种不同类型的语料对其性能的影响。

5.4 语言模型的自适应

处理方法是：将语言模型划分成 n 个子模型 M_1, M_2, …, M_n，整个语言模型的概率通过下面的线性插值公式计算得到：

$$\hat{P}(w_i \mid w_1^{i-1}) = \sum_{j=1}^{n} \lambda_j \hat{P}_{M_j}(w_i \mid w_1^{i-1}) \qquad \ldots(5\text{-}17)$$

其中，$0 \le \lambda_j \le 1$，$\sum_{j=1}^{n} \lambda_j = 1$

λ值可以通过 EM 算法计算出来。

5.4 语言模型的自适应

基本方法：

(1) 对训练语料按来源、主题或类型等聚类(设为*n*类)；

(2) 在模型运行时识别测试语料的主题或主题的集合；

(3) 确定适当的训练语料子集，并利用这些语料建立特定的语言模型；

(4) 利用针对各个语料子集的特定语言模型和线性插值公式(5-17)，获得整个语言模型。

5.4 语言模型的自适应

EM 迭代计算插值系数 λ：

(1)对于n个类，随机初始化插值系数λ；

(2)根据公式(5-17)计算新的概率和期望；

(3)第r次迭代，第j个语言模型在第i ($i \le n$)类上的系数：

$$\lambda_{ij}^{r} = \frac{\lambda_{ij}^{r-1} P_{ij}(w|h)}{\sum_{i=1}^{n} \lambda_{ij}^{r-1} P_{ij}(w|h)}$$

其中，h为历史。

(4)不断迭代，重复步骤(2)和(3)，直至收敛。

5.4 语言模型的自适应

◆ 基于最大熵的语言模型

基本思想：通过结合不同信息源的信息构建一个语言模型。每个信息源提供一组关于模型参数的约束条件，在所有满足约束的模型中，选择熵最大的模型。

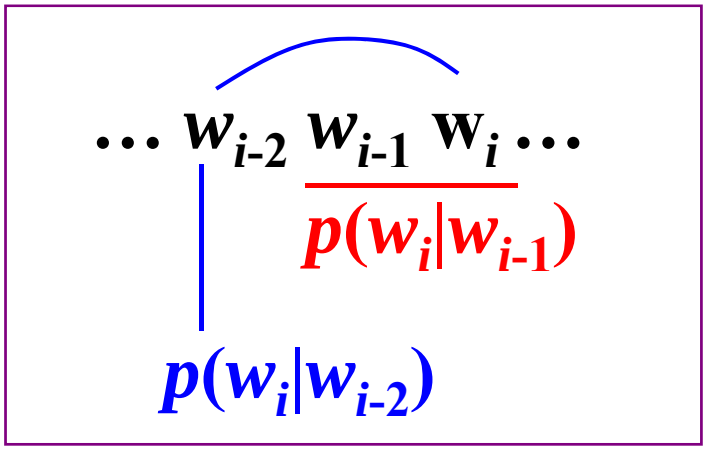
5.4 语言模型的自适应

例如，考虑两个语言模型 M_1 和 M_2，假设 M_1是标准的 2 元模型，表示为 f 函数：

$$\hat{P}_{M_1}(w_i \mid w_1^{i-1}) = f(w_i, w_{i-1}) \qquad \ldots (5\text{-}18)$$

M_2 是距离为2的2元模型 (distance-2 bigram)，定义为g函数：

$$\hat{P}_{M_2}(w_i \mid w_1^{i-1}) = g(w_i, w_{i-2}) \qquad \ldots (5\text{-}19)$$



5.4 语言模型的自适应

用线性插值方法通过取这两个概率估计的平均值，并采用后备(backing-off) 平滑技术来解决这个问题。

最大熵原则将所有的信息源组合成一个模型，对于该模型的约束并不是让公式(5-18)和(5-19)对于所有可能的历史都成立，而是更宽松的限制，即它们在训练数据上平均成立即可，因此，公式(5-18)和(5-19)被分别改写成：

5.4 语言模型的自适应

$$E(\hat{P}_{M_1}(w_i \mid w_1^{i-1}) \mid w_{i-1} = a) = f(w_i, a) \qquad \ldots (5\text{-}20)$$

$$E(\hat{P}_{M_2}(w_i \mid w_1^{i-1}) \mid w_{i-2} = b) = g(w_i, b) \qquad \ldots (5\text{-}21)$$

如果约束条件是一致的(相互之间不矛盾)，那么，总有模型满足这些条件，余下的问题就是利用通用迭代算法 **(generalized iterative scaling, GIS)** 选择使熵最大的模型。

5.5 语言模型应用举例

5.5 语言模型应用举例

- 汉语分词问题：

句子：这篇文章写得太平淡了。

这/ 篇/ 文章/ 写/ 得/ 太/ 平淡/ 了/ 。

这/ 篇/ 文章/ 写/ 得/ 太平/ 淡/ 了/ 。

5.5 语言模型应用举例

采用基于语言模型的分词方法

◆ 方法描述：

设对于待切分的句子$S = z_1 z_2 \ldots z_m$，$W = w_1 w_2 \ldots w_k$ $(1 \le k \le n)$是一种可能的切分。那么，

$$\hat{W} = \arg\max_W P(W \mid S)$$
$$= \arg\max_W P(W) P(S \mid W)$$
$$\cong \arg\max_W P(W)$$

最基本的做法是以词为统计基元，但效果不佳。

5.5 语言模型应用举例

具体实现时，可把汉语词汇分成如下几类：

(1) 分词词典中规定的词；

(2) 由词法规则派生出来的词或短语，如：干干净净、非党员、副部长、全面性、检查员、看不出、克服了、走出来 、洗个澡 …

(3) 与数字相关的实体，如：日期、时间、货币、百分数、温度、长度、面积、重量、电话号码、邮件地址等；

(4) 专用名词，如 ： 人名、地名、组织机构名。

占未登录词的95％！

5.5 语言模型应用举例

进一步做如下约定，把一个可能的词序列 W 转换成词类序列 $C=c_1c_2\cdots c_N$，即：

- 专有名词的人名PN、地名LN、机构名ON分别作为一类；
- 实体名词中的日期 dat、时间tim、百分数per、货币mon 等作为一类；
- 对词法派生词MW和词表词LW，每个词单独作为一类。

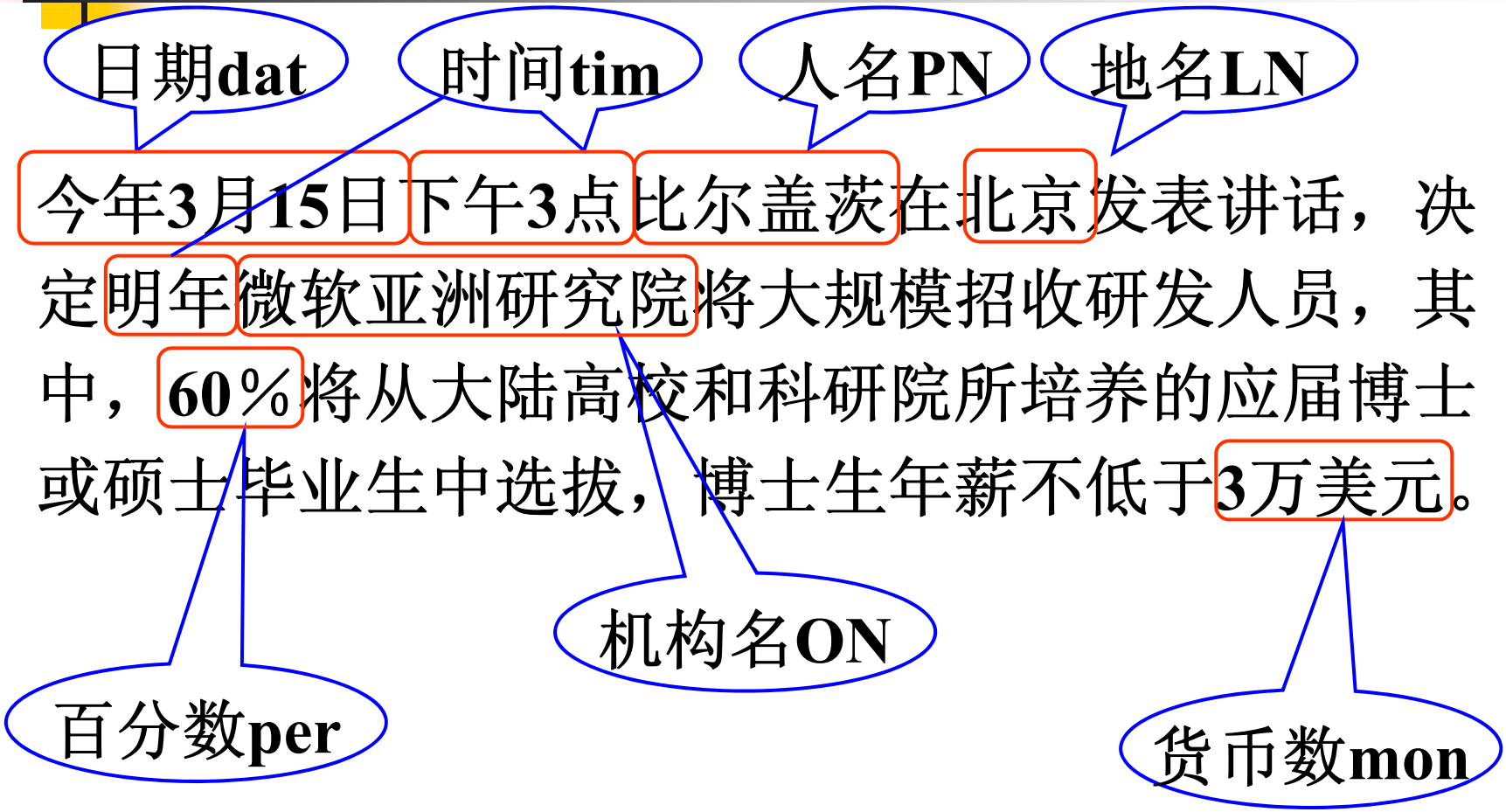
5.5 语言模型应用举例

例如：

今年3月15日下午3点比尔盖茨在北京发表讲话，决定明年微软亚洲研究院将大规模招收研发人员，其中，60％将从大陆高校和科研院所培养的应届博士或硕士毕业生中选拔，博士生年薪不低于3万美元。

Just a joke ☺

5.5 语言模型应用举例


日期dat
时间tim
人名PN
地名LN
今年3月15日下午3点比尔盖茨在北京发表讲话，决定明年微软亚洲研究院将大规模招收研发人员，其中，60％将从大陆高校和科研院所培养的应届博士或硕士毕业生中选拔，博士生年薪不低于3万美元。
机构名ON
百分数per
货币数mon

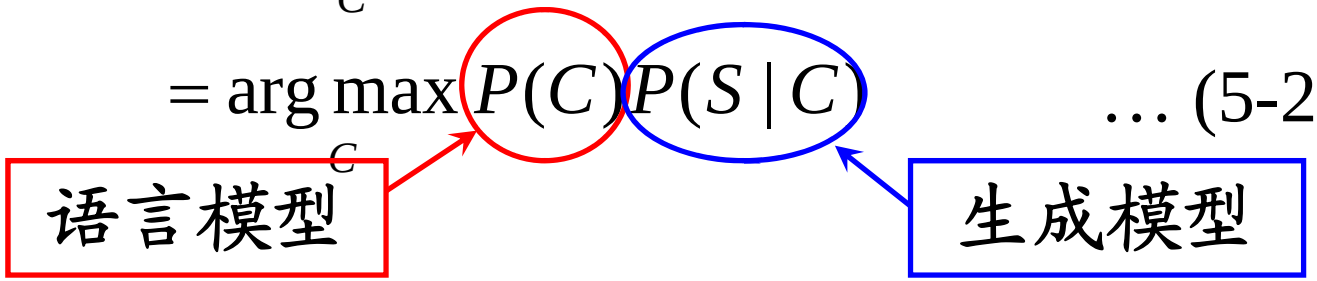
5.5 语言模型应用举例

词序列变为类序列：

dat/ time/ PN/ LN/ 发表/ 讲话/ ，/ 决定/ **tim/ ON/** 将/ 大规模/ 招收/ 研发/ 人员/ ，/ 其中/ ，/ **per/** 将/ 从/ 大陆/ 高校/ 和/ 科研/ 院/ 所/ 培养/ 的/ 应届/ 博士/ 或/ 硕士/ 毕业生/ 中/ 选拔/ ，/ 博士生/ 年薪/ 不/ 低于/ **mon/** 。

5.5 语言模型应用举例

那么，
$$\hat{C} = \arg\max_{C} P(C \mid S)$$
$$= \arg\max_{C} P(C)P(S \mid C) \quad \ldots (5\text{-}22)$$



***P*(*C*)** 可采用三元语法：

$$P(C) = P(c_1)P(c_2 \mid c_1)\prod_{i=3}^{N} P(c_i \mid c_{i-2}c_{i-1}) \quad \ldots (5\text{-}23)$$

$$P(c_i \mid c_{i-2}c_{i-1}) = \frac{count(c_{i-2}c_{i-1}c_i)}{count(c_{i-2}c_{i-1})} \quad \ldots (5\text{-}24)$$

5.5 语言模型应用举例

生成模型在满足独立性假设的条件下，可近似为：

$$P(S \mid C) \approx \prod_{i=1}^{N} P(s_i \mid c_i) \qquad \ldots (5\text{-}25)$$

该公式的含意是，任意一个词类 c_i 生成汉字串 s_i 的概率只与自身有关，而与其上下文无关。例如，如果“教授”是词表里的词，那么

$P(s_i=\text{教授}|c_i=\text{LW})=1$，否则，$P(s_i \mid c_i) = 0$。

5.5 语言模型应用举例

词　类	生成模型 $P(S\|C)$	语言知识来源
词表词 (LW)	若S是词表词，$P(S\|LW)=1$，否则为0;	分词词表
词法派生词 (MW)	若S是派生词，$P(S\|MW)=1$，否则为0;	派生词词表
人名 (PN)	基于字的2元模型	姓氏表，中文人名模板
地名 (LN)	基于字的2元模型	地名表、地名关键词表、地名简称表
机构名 (ON)	基于词类的2元模型	机关名关键词表，机构名简称表
实体名 (FT)	若S可用实体名词规则集G识别，$P(S\|G)=1$，否则为0。	实体名词规则集

5.5 语言模型应用举例

模型的训练由以下三步组成:

(1) 在词表和派生词表的基础上，用一个基本的分词工具切分训练语料，专有名词通过一个专门模块标注，实体名词通过相应的规则和有限状态自动机标注，由此产生一个带词类别标记的初始语料;

(2) 用带词类别标记的初始语料，采用最大似然估计方法估计语言模型的概率参数，公式(5-24);

(3) 用得到的模型（公式(5-22)、(5-23)、(5-25)）对训练语料重新切分和标注，得到新的训练语料;

(4) 重复(2)(3)步，直到系统的性能不再有明显的变化为止。

5.5 语言模型应用举例

实验：

(1)词表词：98,668条、派生词：59,285条；

(2)训练语料：88MB 新闻文本；

(3)测试集：247,039个词次，分别来自描写文、叙述文、说明文、口语等。

指标：

$$\text{正确率 } P = \frac{\text{分词结果中切分正确的 词数}}{\text{分词结果的总词数}} \times 100\% = 96.3\%$$

黄昌宁，高剑峰，李沐，对自动分词的反思，见：2003年全国第七届计算语言学联合学术会议论文集，pp. 26-38

5.5 语言模型应用举例

❑分词与词性标注一体化方法

汉语分词问题：这篇文章写得太平淡了。

这/ 篇/ 文章/ 写/ 得/ 太/ 平淡/ 了/ 。

这/ 篇/ 文章/ 写/ 得/ 太平/ 淡/ 了/ 。

标注词性后：

这/P 篇/M 文章/N 写/V 得/D 太/D 平淡/A 了/X 。/B

5.5 语言模型应用举例

假设句子**S** 是由单词串组成：

$$W = w_1 w_2 \cdots w_n \quad (n \geq 1)$$

单词 $w_i (1 \leq i \leq n)$ 的词性标注为 t_i，即句子**S** 相应的词性标注符号序列可表达为：

$$T = t_1 t_2 \cdots t_n$$

那么，分词与词性标注的任务就是要在**S** 所对应的各种切分和标注形式中，寻找**T**和**W** 的联合概率**P(W, T)** 为最优的词切分和标注组合。

5.5 语言模型应用举例

(1)基于词性的三元统计模型：

$$P(W,T)=P(W|T)P(T)\approx\prod_{i=1}^{n}P(w_i|t_i)P(t_i|t_{i-1}t_{i-2}) \quad \ldots(5\text{-}26)$$

其中，$P(W|T)$ 称为生成模型，$P(w_i|t_i)$ 表示在整个标注语料中，在词性 t_i 的条件下，单词 w_i 出现的概率。$P(T)$ 为基于词性的语言模型，采用三元文法，$P(t_i|t_{i-1}t_{i-2})$ 表示在前两个单词的词性是 t_{i-1} 和 t_{i-2} 的情况下，当前词的词性是 t_i 的概率。

5.5 语言模型应用举例

(2)基于单词的三元统计模型：

$$P(W,T)=P(T|W)P(W)\approx\prod_{i=1}^{n}P(t_i|w_i)P(w_i|w_{i-1}w_{i-2})$$

…(5-27)

其中，$P(t_i|w_i)$反映的是每个词对应词性符号的概率。$P(w_i|w_{i-1},w_{i-2})$是普通的三元语言模型。

5.5 语言模型应用举例

(3)分词与词性标注一体化模型：

$$P^*(W,T)=\alpha\prod_{i=3}^{n}P(w_i\mid t_i)P(t_i\mid t_{i-1},t_{i-2})+\beta\prod_{i=1}^{n}P(t_i\mid w_i)P(w_i\mid w_{i-1},w_{i-2})$$

…(5-28)

这种综合模型的指导思想是希望通过调整参数α和β的值来确定两个子模型在整个分词与词性标注过程中所发挥作用的比重，从而获得分词与词性标注的整体最优。

5.5 语言模型应用举例

从公式 (5-27) 得到的结果分析可知，$P(t_i \mid w_i)$对分词无帮助，且在分词确定后对词性标注又会增添偏差。因此，在实现这一模型时，可以仅取公式 (5-27) 中的语言模型部分，而舍弃词性标注部分，并令 $\alpha = 1$，仅保留加权系统 β，于是，

$$\hat{P}(W,T) = \prod_{i=1}^{n} P(w_i \mid t_i) P(t_i \mid t_{i-1}, t_{i-2}) + \beta \prod_{i=1}^{n} P(w_i \mid w_{i-1}, w_{i-2}) \quad \ldots(5\text{-}29)$$

对比 (5-26) 式，在词性标注方面无改变，但β进一步增强了对分词部分的约束。这样，分词与词性标注一体化问题转化为求解 $\hat{P}(W,T)$最大值的问题。

5.5 语言模型应用举例

在确定β系数值时，可以根据词典中词汇w的个数和词性t的种类数目，取二者之比，即：

β=词典中词w的个数 / 词性t的种类数。

在系统实现时，首先对训练文本进行预处理，将人名、地名和数字串先识别出来，然后用规定的符号分别予以替代，最后再计算相应的条件概率。

5.5 语言模型应用举例

实验：

- **50,000**个常用词的词典
- **13MB**已经切分和标注好的《人民日报》语料训练 $P(w_i \mid t_i)$ 和 $P(t_i \mid t_{i-1}t_{i-2})$
- **110MB**语料训练语言模型 $P(w_i \mid w_{i-1}, w_{i-2})$
- 集内测试集包含**3**个文本，规模分别为：**1,284**、**4,265** 和**9,681**个词
- 集外测试集包含**4**个文本，规模分别为：**719**、**4,644**、**5,627** 和**13,166**个词

5.5 语言模型应用举例

指标：

条件 \ 指标		分词平均正确率(%)	词性标注平均正确率(%)	
			一级词性标注	二级词性标注
使用公式(5-26)	集内	97.78	96.33	93.24
	集外	96.79	96.32	93.10
使用公式(5-28)	集内	99.48	96.28	93.21
	集外	98.06	96.32	93.07

高山，张艳等，基于三元统计模型的汉语分词标注一体化研究，2001年全国第六届计算语言学联合学术会议论文集，pp.116-112

5.5 语言模型应用举例

说明：

- 对于汉语分词而言，不同的分词方法往往各有千秋，不要简单地从正确率高低上判断方法的好坏，正确率只从某一侧面反映了方法的性能；
- 除了汉语自动分词以外，语言模型广泛地应用于自然语言处理的各个方面，是统计自然语言处理方法中最核心、最基本的模型。

本章小结

- *n* 元语法的基本概念
  - uni-gram, bi-gram, tri-gram
- 数据平滑方法：
  - 减值法：1) Good-Turing; 2) Back-off (Katz);
    3) 绝对减值(H. Ney)；4) 线性减值
  - 删除减值法：低阶代替高阶
- 语言模型的自适应方法：
  - Cache-based
  - Hybrid
  - ME-based
- 语言模型的应用

习题

1. 请阅读有关文献，了解除了本讲义介绍的数据平滑方法以外的其它平滑方法；请对 Good-Turing 平滑方法进行简要的评价，阐述你个人的观点。

2. 利用汉语切分和标注语料（注意版权的合法性），尝试用 bi-gram 实现一个简单的汉语自动分词程序。

Thanks