典型参数平滑算法在词性标注中的性能评价*

朱莉　孟遥　赵铁军

哈尔滨工业大学计算机科学与技术学院　哈尔滨 150001

Email:{julie, meng, tjzhao}@mtlab.hit.edu.cn

摘要：随着统计技术在自然语言处理领域的兴起，在语料规模有限的情况下，参数平滑作为解决数据稀疏问题的主要方法显得十分重要。本文分析了几种常用参数平滑算法的优劣，在英语词性标注中比较了这几种算法的平滑效果。实验表明：在语料规模有限的情况下，线性插值和Katz's回退平滑较优。本文通过在相同环境下对各算法的平滑效果的研究和实验，旨在为大家提供一个选择平滑算法的借鉴。

关键字：数据稀疏，参数平滑，词性标注

Comparation of Classical Smoothing Methods in Part-of-Speech Tagging

Zhu Li　Meng Yao　Zhao Tiejun

School of Computer Science & Technology, Harbin Institute of Technology, Harbin, 150001

Email:{julie, meng, tjzhao}@mtlab.hit.edu.cn

Abstract: In statistical natural language processing, parameter smoothing is crucial to solve data parse caused by limited corpus. This paper analyzes several popular smoothing methods and compared their performances in English POS tagging. Experiments indicate that linear interpolation & Katz's perform the best.

KeyWords: Data parse, Parameter smoothing, Part-of-Speech tagging

1　研究背景和意义

随着自然语言处理的研究与发展，在早期的基于规则的自然语言处理方法之后，兴起了基于统计技术的自然语言处理方法，即人们通过对标准的语料库进行一定需要的学习，从中获取处理所需知识的方法。

其中，语料库(Corpus)是按照一定原则组织在一起的真实自然语言数据(包括书面语和口语)集合，主要用于研究自然语言的规律，特别是统计语言学模型的训练以及相关系统的评价和评测[1]。由于语料库的规模和它所包含的语言现象有限，从而导致数据稀疏现象的产生，这是制约基于统计的自然语言处理方法发展的重要因素。数据稀疏现象是指，在基于统计技术的知识获取方法中，在语料库的规模不够大的条件下，大多数词或邻接词及各属性的搭配在语料中出现的次数很少，甚至根本不出现的现象。即许多合法的、在未来的文本中可能要遇到的标记在统计语料中出现的次数很小或从未出现过，从而造成知识短缺的现象[2]。在实

*本文研究受到国家863计划资助(项目号2002AA117010-09)。

际研究中，数据稀疏的存在会产生大量空值，严重影响后续处理的性能和效果。为解决数据稀疏带来的问题，大家很自然的想到扩大语料库的规模，而语料库的建设代价高，并且由 Zipf 定律[3]知扩大语料库对数据稀疏问题的解决是有限的。因此在有限的语料规模下，研究参数空间的平滑算法就十分重要了。参数平滑即在训练数据不足够充分的条件下，采用某种方式对统计结果和概率估计进行必要的调整和修补，以降低由于数据稀疏现象带来的统计误差。

随着自然语言处理对精度的要求越来越高，人们根据研究需要建立了许多解决数据稀疏问题的参数平滑算法，为我们在实际的研究过程中提供了多种选择。但每种方法又有着各自的优势，因此各种算法的平滑效果到底如何，在什么情况下适合选择什么样的平滑算法，成为困扰大家的问题。本文通过在相同环境下对不同平滑算法的平滑效果的研究和实验，剖析了几种典型平滑算法的优缺点，旨在为大家提供一个选择平滑算法的依据。

2 平滑算法概述

随着统计技术在自然语言处理领域的提出和日益广泛的应用，有限的语料库带来的数据稀疏问题成为影响自然语言处理效果的主要因素之一。于是研究者们根据研究的需要建立了各种参数平滑算法。

需要指出的是，所有参数平滑算法都基于实际统计的数据之上，是对实际统计数据的平滑，大多数情况下我们采用极大似然估计来统计实际数据，对数据的平滑自然是在此数据基础上进行的。根据平滑过程中实际统计数据的改变情况，平滑算法可以分成两大类：一类是修改所有实际统计数据的参数平滑算法；另一类是修改部分实际统计数据的参数平滑算法。

修改所有实际统计数据的参数平滑算法主要有：加法平滑[4][5](Additive Smoothing)、Good-Turing 估计[6]、折扣参数平滑(Discounting Smoothing)、线性插值平滑(Linear Interpolation Smoothing)、基于扣留估计的参数平滑技术[7]。加法平滑 (Additive Smoothing)，由 Lidstone、Johnson 和 Jeffreys 等人提出，是一种简单易行的数据平滑方法[8]。该方法的基本思想是：为了避免零概率问题，为每个实际统计的数据加一个常量。该常量是个经验值，经常取 1 和 0.5。这种平滑技术的性能一般来说较差[9]。Good-Turing 估计基本思想是：将统计参数按出现次数聚类，用出现次数加 1 的类来估计当前类。折扣参数平滑(Discounting Smoothing)，主要包括绝对折扣参数平滑(Absolute Discounting Smoothing)和线性折扣参数平滑(Linear Discounting Smoothing)，基本思想：按照某种比例从观察到的参数数据中折扣出一部分平分给未观察到的参数。如果折扣参数选取的好，绝对折扣参数平滑能取得比较好的效果[7]，但线性折扣参数平滑很难评价 [7]。线性插值平滑(Linear Interpolation Smoothing)的基本思想是：利用低元参数的线性组合来估计高元参数，应用范围广泛。基于扣留估计的参数平滑技术，主要包括扣留估计参数平滑(Held-out Estimation Smoothing)和交叉校验参数平滑(Cross-validation Smoothing)。基本思想是给语料分块，利用语料块间的差异来平滑参数空间。

修改部分实际统计数据的参数平滑算法主要有：给定最小值平滑(Clipping with a floor Value)，基本思想是对所有出现次数或概率值为 0 的参数给定一个 floor value，该值的选取依经验而定。Katz's 式平滑[10]，基本思想：保留部分极大似然估计的概率值，利用 Good-Turing 修改出现次数少的参数的概率值，对于概率值为 0 的参数采用回退的方法来估计。

各种平滑算法中，以 Good-Turing 估计、线性插值平滑(Linear Interpolation Smoothing)、

交叉校验参数平滑(Cross-validation Smoothing)、Katz's 回退式平滑最为典型和常用，本文在英语词性标注中对比各平滑算法的优劣，以期在英语词性标注中选择最适合的参数平滑算法。同时在相同的实验条件下，平滑算法的对比结果也可以为研究人员选择平滑算法提供借鉴。

3 典型平滑算法分析

3.1 Good-Turing 估计

Good-Turing 估计是许多数据平滑技术的核心[8]。该方法的基本思想是：将统计参数按出现次数聚类，用出现次数加 1 的类来估计当前类的概率。

对于 N-gram 模型中出现 r 次的 N 元对 w_{i-n+1}^{i}，根据 Good-Turing 估计公式，该 N 元对的出现次数为 r^*：$r^* = (r+1)\frac{E(n_{r+1})}{E(n_r)}$，出现概率为：$p_{GT}(w_{i-n+1}^{i}) = \frac{r^*}{\sum_{r=0}^{\infty} r^*}$

其中 n_r 是训练集中实际出现 r 次的 N 元对的个数，$E(n_r)$ 是 n_r 的期望，在算法实现的过程中可以用 n_r 来代替 $E(n_r)$，对于比较大的 r，用观察值替代期望值是不可靠的。

Good-Turing 估计适合单词量大并具有大量的观察数据的情况下使用，在观察数据不足的情况下，本身出现次数就是不可靠的，利用它来估计出现次数就更不可靠了；另外，由于上面的公式中缺乏利用低元模型对高元模型进行线性插值的思想，它通常不单独使用，而作为其他平滑算法中的一个计算工具。

3.2 线性插值平滑(Linear Interpolation Smoothing)

线性插值平滑（Linear Interpolation Smoothing）方法通常也被称作 Jelinek-Mercer 平滑[11]。Jelinek 和 Mercer 在 1980 年首先提出了这种数据平滑算法的思想，Brown 在 1992 年给出了线性插值的平滑公式：

$$p_{\text{int}erp}(w_i \mid w_{i-n+1}^{i-1}) = \lambda_{w_{i-n+1}^{i-1}} \cdot p_{ML}(w_i \mid w_{i-n+1}^{i-1}) + (1-\lambda_{w_{i-n+1}^{i-1}}) \cdot p_{\text{int}erp}(w_i \mid w_{i-n+2}^{i-1})$$

对于插值系数 $\lambda_{w_{i-n+1}^{i-1}}$ 的估计，一般采用 Baum-Welch 算法[12]估计。使用经过数据平滑的模型，计算一个测试集 H 的对数似然概率 $\log p(H)$，当 $\log p(H)$ 为极大值时，对应的 $\lambda_{w_{i-n+1}^{i-1}}$ 为最优值。因此可以通过对方程 $\frac{\partial \log p(H)}{\partial \lambda_{w_{i-n+1}^{i-1}}} = 0$ 求解，得到迭代计算公式：

$$\lambda'_{w_{i-n+1}^{i-1}} = \frac{1}{c(w_{i-n+1}^{i-1})} \cdot \sum_{w_i} c(w_{i-n+1}^{i}) \cdot \frac{\lambda_{w_{i-n+1}^{i-1}} \cdot p_{ML}(w_i \mid w_{i-n+1}^{i-1})}{\lambda_{w_{i-n+1}^{i-1}} \cdot p_{ML}(w_i \mid w_{i-n+1}^{i-1}) + (1-\lambda_{w_{i-n+1}^{i-1}}) \cdot p_{\text{int}erp}(w_i \mid w_{i-n+2}^{i-1})}$$

其中，$c(w_i^j)$ 代表词串 w_i^j 在测试集中出现的次数。

该参数平滑技术的基本思想是利用低元 n-gram 模型对高元 n-gram 模型进行线性插值。用降元的方法来弥补高元的数据稀疏问题，数据估计有一定的可靠性。但是参数估计较困难，

在实际应用需采用简化方法。本文采用[13]中提到的参数确定方法。

3.3 Katz's 式平滑

Katz's 式平滑是 Katz 提出的一种回退式数据平滑算法[10]，该算法的基本思想是：当一个 N 元对 w_{i-n+1}^{i} 的出现次数 $c(w_{i-n+1}^{i})$ 足够大时，$p_{ML}(w_i \mid w_{i-n+1}^{i-1})$ 是 w_{i-n+1}^{i} 可靠的概率估计。而当 $c(w_{i-n+1}^{i})$ 不足够大时，采用 Good-Turing 估计对其平滑，将其部分概率折扣给未出现的 N 元对。当 $c(w_{i-n+1}^{i})=0$ 时，模型回退到低元模型，按着 $p_{katz}(w_i \mid w_{i-n+2}^{i-1})$ 比例来分配被折扣给未出现的 N 元对的概率。综合上述思想，回退式平滑算法的平滑公式为：

$$p_{katz}(w_i \mid w_{i-n+1}^{i-1}) = \begin{cases} p_{ML}(w_i \mid w_{i-n+1}^{i-1}) & if(c(w_{i-n+1}^{i}) \geq k) \\ \alpha \cdot p_{GT}(w_i \mid w_{i-n+1}^{i-1}) & if(1 \leq c(w_{i-n+1}^{i}) < k) \\ \beta \cdot p_{katz}(w_i \mid w_{i-n+2}^{i-1}) & if(c(w_{i-n+1}^{i}) = 0) \end{cases}$$

其中 $p_{ML}(w_i \mid w_{i-n+1}^{i-1})$ 为最大似然估计模型。$p_{GT}(w_i \mid w_{i-n+1}^{i-1})$ 为 Good-Turing 概率估计。阈值 k 为一个常量，Katz 建议 k=5。参数 α 和 β 保证模型参数概率的归一化约束条件，即 $\sum_{w_i} p_{katz}(w_i \mid w_{i-n+1}^{i-1}) = 1$。为了结束公式的递归定义，可以令一元模型为最大似然估计模型。

于是可以得到：$\alpha = \dfrac{\sum_{r=2}^{K-1} r \cdot n_r}{\sum_{r=2}^{K} r \cdot n_r}$，其中 $r = c(w_{i-n+1}^{i})$，$n_r = \sum_{w_{i-n+1}^{i}} \delta(c(w_{i-n+1}^{i}), r)$

$$\beta = \frac{1 - \alpha \cdot \sum_{w_3 \in S_K} p_{GT}(w_i \mid w_{i-n+1}^{i-1}) - \sum_{w_3 \in S_{K'}} p_{ML}(w_i \mid w_{i-n+1}^{i-1})}{\sum_{w_3 \in S_0} p_{katz}(w_i \mid w_{i-n+2}^{i-1})}$$

其中 $N = \sum_{w_{i-n+1}^{i}} c(w_{i-n+1}^{i})$，$p_{GT}(w_{i-n+1}^{i}) = \dfrac{n_{r+1}}{n_r} \cdot \dfrac{r+1}{N}$，$p_{ML}(w_{i-n+1}^{i}) = \dfrac{r}{N}$，$S_0 = \{w_i \mid c(w_{i-n+1}^{i}) = 0\}$，$S_K = \{w_i \mid 0 < c(w_{i-n+1}^{i}) < K\}$，$S_{K'} = \{w_i \mid c(w_{i-n+1}^{i}) \geq K\}$。与线性插值平滑算法相比，回退式数据平滑算法的参数较少，而且可以直接确定，无需通过某种迭带重估算法反复训练，因此它的实现更为方便[14]。从该算法的计算公式中我们就可以看出，它既保留了与实际数据分布较接近的大部分数据、对于出现次数较少的 N 元组采用 Good-Turing 的方法进行估计，对统计概率值为 0 的 N 元组又体现了低元估计高元的思想。

3.4 交叉校验参数平滑(Cross-validation Smoothing)

前面介绍的方法都是将训练样本看成一个整体，然后在实际统计数据上进行平滑。而交叉校验参数平滑是基于扣留技术的一种平滑方法，它是将训练样本分成若干个小的部分，通

过各部分之间的数据差异来实现对参数的平滑。

下面将训练语料分成两部分(部分 0 和部分 1，将部分 1 作为扣留语料)来说明该方法的操作过程。$C_i(w_n^i)$ 表示 w_n^1 在部分 1 中出现的次数，N_r^0 表示部分 0 中出现次数为 r 的 n-gram 的个数，N_r^1 表示部分 1 中出现次数为 r 的 n-gram 的个数。$T_r^{ab} = \sum_{\{w_n^1:C_a(w_n^1)=r\}} C_b(w_n^1)$ 表示在部分 a 中出现 r 次的 n-gram 在部分 b 中出现的次数和，于是有交叉校验参数平滑公式：

$$P(w_n^1) = \frac{T_r^{01} + T_r^{10}}{N(N_r^0 + N_r^1)}, \quad C(w_n^1) = r。$$

该方法采用将训练语料分块的方法，利用数据块的统计差异来平滑各语料块中参数的概率。显然，对语料的依赖性很大，平滑效果受语料分块方法的限制；并且该方法对于各块语料中都未出现的 N 元组的概率值是无法估计的，平滑效果很有限。

4 实验结果与分析

本文将上述平滑算法分别应用于基于三元 HMM 的英语的词性标注实验中，以三元 HMM 的词性标注为实验平台，测试上述典型平滑算法的性能。需要指出的是：本文旨在比较各平滑算法的平滑效果，实验中未考虑未登录词和词法分析的问题，故实验结果并没有达到其他有关词性标注文献中提到的最好结果。实验选取的语料是 Penn TreeBank 语料库。实验从该语料库中随机的抽取了 2220 句(470KB)作为所有平滑算法的开放测试语料，另外 44000 句(9.01M)作为训练语料，并从前 11000 句中随机抽取 2000(409KB)句作为封闭测试的语料。

实验一：语料分块的不同对交叉校验平滑算法的影响

该实验是为了比较交叉校验平滑算法中语料的分块比例对平滑效果的影响，对于 Penn TreeBank 的 44000 句(9.01M)训练语料我们做了 4 次实验，分别将语料按不同比例(1:1、2:1、3:1、4:1)划分成两部分 part0 和 part1 进行参数空间的平滑，测试结果如下：

表一：语料分块的不同对交叉校验平滑算法的影响

实验语料比例(part0:part1)	封闭测试(2000 句)%	开放测试(2220 句)%
1:1	95.90	93.68
2:1	95.21	93.11
3:1	94.26	92.28
4:1	93.24	91.64

从实验结果可以看出：当语料块大小接近时，交叉校验的平滑效果较好，这一点是可以解释的。当语料分块不均匀时必然有个别语料块的规模较小，在这块语料上对参数进行估计显然会与实际参数的分布相差较大，如果利用这块语料上的参数统计与其他语料块上的参数统计进行交叉校验，不但不能更好的平滑参数空间反而降低了其他语料块对参数的估计，只有当语料分块的均匀度比较高时，各块上的参数统计才比较接近，平滑效果才会随之提高；

另外交叉校验参数平滑对在语料中未出现的参数不做任何平滑，导致它的平滑效果不是很好。总的来说交叉校验平滑高度依赖语料规模和分块情况，平滑效果并不理想。

实验二：英语词性标注中不同平滑函数的性能比较

在该实验中，每次增加 11000 句测试语料，利用各种平滑方法对参数进行估计，其中交叉校验平滑方法中分块比例为 1:1。实验测试结果见下页：

从表中数据可以看出：(1)随着语料库规模的不断增大，开放测试中各参数平滑算法的平滑效果都有所提高，而封闭测试中，正确率有少量的降低。这一点很显然，当语料规模增大时，对参数的统计更接近实际分布的情况，在此基础上进行的平滑效果也会相应提高，开放测试正确率呈上升趋势；语料库规模越小，统计数据与训练语料中的概率分布越接近，封闭测试结果也越好，所以当语料规模扩大时封闭测试结果反而有所下降。另外，还可以看到，语料规模对封闭测试结果比对开放测试结果的影响小；(2)对于各种规模的语料库以线性插值

表二：英语词性标注中不同平滑函数的性能比较

训练语料		封闭测试(2000 句)%					开放测试(2220 句)%				
句子	单词	不平滑	Good	交叉	线性	Katz's	不平滑	Good	交叉	线性	Katz's
11000	260305	96.06	94.68	95.63	96.22	96.20	91.69	89.95	91.54	93.62	93.81
22000	520038	95.96	94.61	96.04	96.19	96.21	92.72	90.21	91.55	94.39	94.15
33000	778020	95.90	94.57	95.93	96.15	96.13	93.57	91.11	92.26	95.11	95.03
44000	1037724	95.84	94.69	95.90	96.13	96.11	94.01	92.54	93.68	95.23	95.12

平滑和 Katz's 回退平滑的效果为最佳。这两种平滑方法都采用了低元估计高元的思想，区别在于线性插值平滑它改变了所有的统计数据，用高低元的线性组合来平滑参数，是一种类似基于记忆的方法；而 Katz's 回退平滑则保留了认为较可靠的统计数据(出现次数达于到一定数目的参数)，完全相信语料，对于出现次数较少的参数采用 Good-Turing 的方法进行估计，对于未出现的参数采用低元估计高元的方法。从计算公式上就可以看出，这两种平滑方法更尊重原始语料，因而取得了较好的平滑效果；(3)Good-Turing 和交叉校验平滑的平滑效果不如不平滑方法。由 Good-Turing 估计的计算公式就可以看出：它是用参数出现次数来进行估计的，对于出现次数较小的参数适当的提高概率是合适的，但当出现次数较大时再进行这样的平滑概率值的提高就有些过量，平滑效果不理想；(4)随着语料规模的不断增大，各平滑方法的平滑差距在减小。当语料库规模足够大的时候，可以认为统计概率分布基本上是接近实际的概率分布的。在此基础上进行的平滑也会更接近真实概率分布，各平滑方法的差距也不断的减小；(5)线性插值和 Katz's 回退平滑在语料库规模较小的情况下也可以有较好效果，实验表明：训练语料 11000 句时的效果就可以接近不平滑时使用全部 44000 句时获得的效果。也从一个侧面说明，在统计自然语言处理中，适当的平滑函数可以更好的模拟实际参数的概率分布，从而有效降低统计算法对训练语料的依赖。实验表明在有限的语料规模下通过合理的参数平滑可以获得比较理想的结果。

5 结论

随着基于统计技术的自然语言处理方法的兴起，在语料规模有限的情况下，参数平滑作

为解决数据稀疏问题的主要方法就显得十分重要。本文首先概述了各种参数平滑方法，并以是否改变统计数据为标准对其进行了分类。随后选择了四种最常用和典型的参数平滑算法进行了细致的剖析，并在三元 HMM 的词性标注中，对其进行了比较研究。实验显示与不平滑的词性标注方法相比，采用适当的平滑算法可以有效地提高词性标注正确率。对比实验表明：在语料有限的情况下，采用何种平滑算法，对实验效果影响很大。四种平滑算法的比较表明，线性插值参数平滑和 Katz's 回退参数平滑的效果较好；Good-Turing 估计只适合对低频参数进行平滑不适合对高频参数进行平滑；交叉校验参数平滑高度依赖语料规模和语料分块，并且它没有提供对未出现参数进行估计的方法，其平滑效果最不理想。本文通过在相同环境下对不同平滑算法的平滑效果的研究和实验，为大家选择平滑算法提供了借鉴。

参 考 文 献

[1] R. A. Cole, J. Mariani, H. Uszkoreit, A. Zaenen, and V. Zue.: "Survey of the State of the Art in Human Language Technology", http://cslu.cse.ogi.edu/HLTsurvey/

[2] 江铭虎 朱小燕 袁保宗: "一种适应域的汉语 N-gram 语言模型平滑算法"，清华大学学报自然科学版 (Journal of TSINGHUA University Science and Technology) 1999 年 第 39 卷 第 9 期 Vol.39 No.9 1999 http://power.luncng.com/power/librarv/qhdxxb/qhdx99/qhdx9909/990927.htm

[3] G. Zipf.: "The Psycho-Biology of Language" Houghton. Mifflin.

[4] Fienberg, S. E. and P. W. Holland: "On the Choice of Flattening Constants for Estimating Multinomial Probabilities", Journal of Multivariate Analysis, Vol. 2, pp. 127-134, 1972

[5] Su., K.-Y., M.-H. Su and L.-M. Kuan.: "Smoothing Statistic Databases in a Machine Translation System", Proceedings of ROCLING-II, pp. 333-347, Academic Sinica, Taipei, Taiwan, R.O.C., Sept. 22-24, 1989

[6] Good, I. J.: "The population frequencies of species and the estimation of population parameters", Biometrika, vol. 40, n0. 3, 4, pp. 237-264, 1953.

[7] Christopher D. Manning, Hinrich Schutze: "Foundations of Statistical Natural Language Processing", The MIT Press. Cambridge, Massachusetts, London, England

[8] H. Jeffreys.: "Theory of Probability", Clarendon Press, Oxford, Second Edition, 1948

[9] W. A. Gale and K. W. Church: "Estimation Procedures for Language Context: Poor Estimates Are Worse Than None", Proceedings in Computational Statistics, Ninth Symposium, pages 69-74, Dubrovnik, Yugoslavia, September 1990.

[10] Katz, S. M.: "Estimation of Probabilities From Sparse Data for the Language Model Component of a Speech Recognizer", IEEE Transactions on Acoustics, Speech and Signal Processing, No. 35, pp. 400-401, 1987.

[11] F. Jelinek and R. L. Mercer: "Interpolated Estimation of Markov Source Parameters from Sparse Data", In Proceedings of the Workshop on Pattern Recognition in Practice, Amsterdam, Netherlands: North-Holland, May 1980.

[12] A. B. Poritz.: "Hidden Markov Models: A Guided Tour. Proceedings of the International Conference on Acoustics", Speech, and Signal Processing, 1: 7-13, New York Hilton, New York City, April 1988.

[13] Scott M. Thede & Mary P. Harper: "A Second-Order Hidden Markov Model for Part-of-Speech Tagging" 1999

[14] 徐志明: "面向文字识别的汉语统计语言模型研究" 2001 博士论文 43-44