<Journal of Contemporary Linguistics>, 2005, No. 2.

Probabilistic Grammar in Natural Language Processing (revised)

Feng Zhiwei

Korterm, KAIST, Daejeon, Korea
Ministry of Education, Institute of Applied Linguistics, Beijing, China
e-mail: zwfeng@world.kaist.ac.kr
zwfengde@public.bta.net.cn

Abstract: Probabilistic grammar assigns a probability to a sentence or a string of words, while attempting to capture more sophisticated syntactic information than the context-free grammar (CFG). A probabilistic context-free grammar (PCFG) is a context-free grammar in which every rule is annotated with the probability of choosing that rule. Each PCFG rule is treated as if it were conditionally independent; thus the probability of a sentence is computed by multiplying the probabilities of each rule in the parse of the sentence. The CYK algorithm is a bottom-up dynamic programming parsing algorithm. It can be augmented to compute the probability of a parse while it is parsing a sentence. PCFG probabilities can be learned by counting in a parsed corpus (tree-bank), or by parsing a corpus. The Inside-Outside algorithm is a way of dealing with the fact that the sentence being parsed is ambiguous. Probabilistic lexicalized context-free grammar augments PCFG with a lexical head for each rule. The probability of a rule can be conditioned on the lexical head or nearby head.

Keyword: context-free grammar, probabilistic context-free grammar, probabilistic CYK algorithm, probabilistic lexicalized context-free grammar

《当代语言学》，2005 年，第 2 期

自然语言处理中的概率语法（修改稿）

冯志伟

韩国科学技术院计算机系, Korterm, KAIST
教育部语言文字应用研究所

摘要　上下文无关语法在处理句法歧义时遇到了困难，概率上下文无关语法和概率词汇化上下文无关语法为解决句法歧义问题提供了有力的手段。概率语法给一个句子或者单词的符号串指派一个概率，从而捕捉比一般的上下文无关语法更加细致的句法信息。概率上下文无关语法也是一种上下文无关语法，其中的每一个规则都标上选择该规则的概率，处理每一个上下文无关规则时，都假定它们在条件上是独立的，一个句子的概率使用剖析该句子时每一个规则的概率的乘积来计算。CYK 算法是一种自底向上的动态规划剖析算法，经过概率上下文无关语法增强之后，概率 CYK 算法就可以计算在剖析一个句子时的剖析概率。概率上下文无关语法的概率可以通过统计已经剖析好的语料库（树库）中的规则而得到，也可以通过直接剖析尚未加工过语料库而得到。当剖析的句子有歧义的时候，可以使用向内向外算法来处理。概率词汇化上下文无关语法要考虑每一个规则的词汇中心语，规则的概率要以词汇中心语和邻近的中心语作为它的条件，这样就进一步增强了规则的功能。

关键词　上下文无关语法　概率上下文无关语法　概率 CYK 算法　概率词汇化上下文无关语法

在自然语言的计算机处理中，基于规则的句法剖析主要是使用 Chomsky 的上下文无关语法。在上下文无关语法的基础上，学者们提出了自顶向下分析法、自底向上分析法、左角分析法、CYK 算法、Earley 算法、线图分析法等行之有效的剖析技术（冯志伟，2000, 2002, 2003a, 2003b）。但是，这些分析方法在处理自然语言的歧义时都显得无能为力。近年来对上下文无关语法的改进主要体现在两个方面：一方面是给上下文无关语法的规则加上概率，提出了概率上下文无关语法，另一方面是除了给规则加概率之外，还考虑规则的中心词对于规则概率的影响，提出了概率词汇化上下文无关语法。这些研究把基于规则的理性主义方法与基于统计的经验主义方法巧妙地结合起来，取得了很好的成果，反映了当前自然语言处理的新趋势。本文主要介绍概率上下文无关语法和概率词汇化上下文无关语法，我们把它们统称为自然语言处理中的概率语法。

1. 上下文无关语法与句子的歧义

上下文无关语法（Context-Free Grammar, 简称 CFG）G 可以定义为四元组 G={N, Σ, P, S}。其中，N 是非终极符号的集合，Σ是终极符号的集合，S 是初始符号，P 是重写规则，规则的形式为

$$A \rightarrow \beta$$

规则左部的 A 是单独的非终极符号，规则的右部β是符号串，它可以由终极符号组成，也可以由非终极符号组成，还可以由终极符号和非终极符号混合组成。

例如，我们有如下的上下文无关语法{N, Σ, P, S}：

N = {S, NP, VP, PP, Prep, Verb, Noun}

Σ = {like, swat, flies, ants}

S = {S}

P:

S → NP VP

S → VP

NP → Noun

NP → Noun PP

NP → Noun NP

VP → Verb

VP → Verb NP

VP → Verb PP

VP → Verb NP PP

PP → Prep NP

Prep → like（含义是“如像”）

Verb → swat (含义是“猛击”)

Verb → flies (含义是“飞”，单数第三人称现在时)

Verb → likes（含义是“喜欢”）

Noun → swat (专有名词，苍蝇的名字)

Noun → flies (含义是“苍蝇”，复数)

Noun → ants （含义是“蚂蚁”，复数）

我们注意到，swat 可以做动词使用，也可以做专有名词使用；likes 可以做动词使用，也可以做介词使用；flies 可以做动词使用，也可以做名词使用。

如果我们使用上下文无关语法的剖析技术（如线图分析法、Earley 算法等），根据这样的规则来剖析英语句子“swat flies like ants”，可以得到如下三个结构不同的树形图：

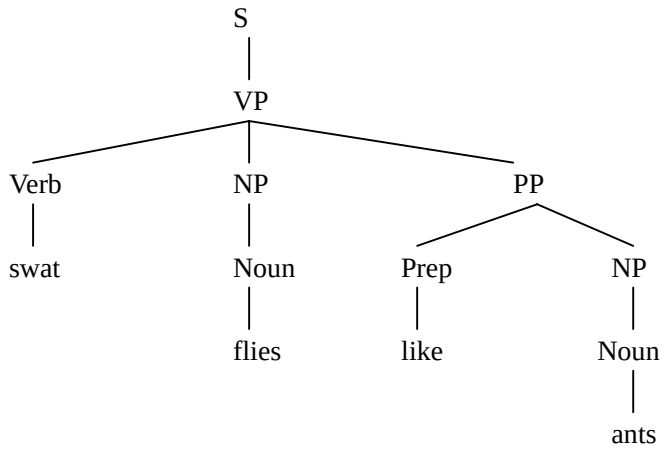


图 1. 树形图 T1

具有这个树形图结构 T1 的句子的含义是“像猛击蚂蚁一样地猛击苍蝇”。

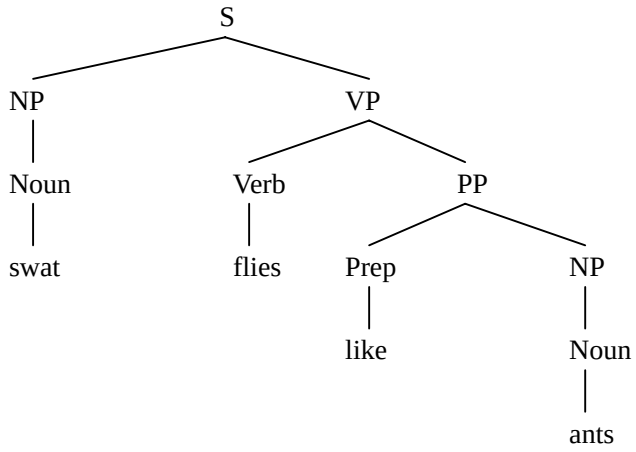


图 2. 树形图 T2

具有这个树形图结构 T2 的句子的含义是“Swat 像蚂蚁一样地飞”。

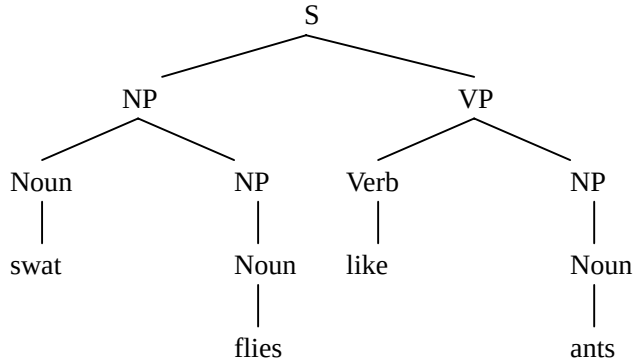


图 3. 树形图 T3

具有这个树形图结构 T3 的句子的含义是“叫做 Swat 的一些苍蝇喜欢蚂蚁”。

同样一个英语句子得到了三种不同的分析结果。究竟这个句子的结构和含义是什么？我们处于举棋不定、进退两难的困境。

目前已经提出了不少基于规则的歧义消解方法来排除歧义，例如，基于选择限制的方法、基于词典的词义排歧方法等。但是这些基于规则的方法消解歧义的效果都不很理想。于是，学者们试图改进上下文无关语法，采用基于统计的方法，计算上下文无关语法重写规则的使用概率，试图根据概率来改进上下文无关语法。

在自然语言处理中关于规则方法和统计方法的争论反映了语言学中的理性主义思潮与经验主义思潮的对立。有一些学者往往持相当极端的观点。

Noam Chomsky 早在 1956 年就说，，“然而应当认识到，‘句子的概率’这个概念，在任何已知的对于这个术语的解释中，都是一个完全无用的概念。” Chomsky 完全无视“句子的概率”，他对于统计方法是嗤之以鼻的。

而 IBM 公司 Watson 研究中心语音研究组的负责人 Fred Jelinek 在 1998 年（当时他在 IBM 语音研究组）却完全排斥规则方法，他竟然语出惊人地说：“每当语言学家离开我们的研究组，语音识别率就提高一步 [1]。”他试图完全排斥熟悉语言规则的语言学家参加自然语

[1] 这是 Jelinek 于 1988 年 12 月 7 日在自然语言处理评测讨论会上的讲话。在 Palmer 和 Finin（1990）描述这个讨论会时，没有写下这段引文；一些当时参加会议的人回忆，Jelinek 讲的话更为尖刻，他说：“Every time I fire a linguist the performance of the recognizer improves.”（“每当我解雇一个语言学家，语音识别系统的性能就会改善一些。”）

言处理的研究。

应该说，这两位著名学者的意见都是非常偏颇的。

更多的学者则以平和的心态，积极地探索把规则方法和统计方法相互结合的途径，他们的研究主要包括两方面，一是提出概率上下文无关语法，二是提出概率词汇化上下文无关语法。下面我们分别介绍这两种语法。

2. 概率上下文无关语法

概率上下文无关语法(Probabilistic Context-Free grammar，简称 PCFG)又叫做随机上下文无关语法（Stochastic Context-Free Grammar ，简称 SCFG)。这种语法是由 Booth（1969）最早提出来的。

上下文无关语法可以定义为四元组{N, Σ, P, S}。而概率上下文无关语法则在每一个重写规则 A→ β 上增加一个条件概率 p ：

A→ β [p]

这样。上下文无关语法就可定义为一个五元组 G = {N, Σ, P, S, D}，其中 D 是给每一个规则指派概率 p 的函数。这个函数表示对于某个非终极符号 A 重写为符号串 β 时的概率 p。这个规则可写为：

P（A→ β）

或者写为：

P (A→ β | A)

从一个非终极符号 A 重写为 β 时应该考虑一切可能的情况，并且其概率之和应该等于 1。

例如，根据对于语料库中规则出现概率的统计，我们可以获得规则的概率，这样，我们就可以在前面的那个上下文无关语法的规则中，给每一条规则加上概率了。这样，我们也就可以把前面的上下文无关语法改进为一个包含概率规则的上下文无关语法了。这些包含概率的规则如下：

S → NP VP	[0.8]
S → VP	[0.2]
NP → Noun	[0.4]
NP → Noun PP	[0.4]
NP → Noun NP	[0.2]
VP → Verb	[0.3]
VP → Verb NP	[0.3]
VP → Verb PP	[0.2]
VP → Verb NP PP	[0.2]
PP → Prep NP	[1.0]
Prep → like	[1.0]
Verb → swat	[0.2]
Verb → flies	[0.4]
Verb → likes	[0.4]
Noun → swat	[0.05]
Noun → flies	[0.45]
Noun → ants	[0.05]

注意，这些规则中，所有从同一个非终极符号重写的规则的概率之和为都为 1。只有以

Noun 为左部的规则的概率之和不为 1，由于名词数量，我们只简单地列举了几条。这些数据来自 Eugene Charniak 的专著《统计语言学习》(Statistical Language Learning, The MIT Press, 1993.)，都是示例性的。准确的数据应该到树库中去获取。

如果分析的句子是有歧义的，概率上下文无关语法可给句子的每一个树形图一个概率。一个树形图 T 的概率应该等于从每一个非终极符号的结点 n 扩充的规则 r 的概率的乘积:

$$P(T) = \prod_{n \in T} p(r(n))$$

其中，n 表示非终极符号的结点，r 表示由该非终极符号扩充的规则，小写字母 p 表示规则 r 的概率，T 表示树形图，大写字母 P 表示整个树形图的概率。这样一来，就可以比较不同树形图的概率，从而进行歧义的消解了。

例如，我们可以在前面那个句子“swat flies like ants”的三个不同的树形图的每一个非终极结点上，加上相应规则的概率。

树形图 T1 加了概率之后变为：

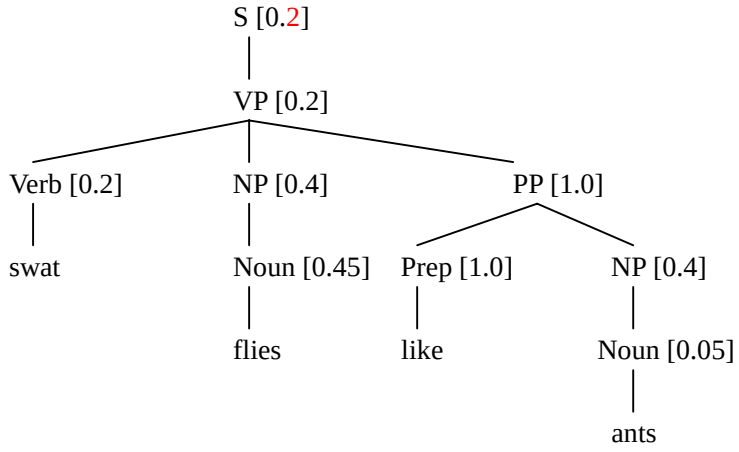


图 4. 非终极结点上加了概率的树形图 T1

把结点上的相应规则的概率相乘，就可以计算出树形图 T1 的概率如下：

$$P(T1) = 0.2 \times 0.2 \times 0.2 \times 0.4 \times 0.45 \times 1.0 \times 1.0 \times 0.4 \times 0.05$$
$$= 2.88 \times 10^{-5}$$

树形图 T2 加了概率之后变为：

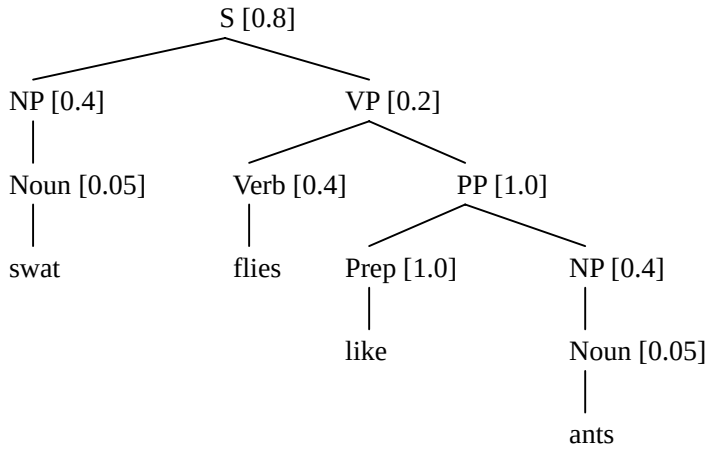


图 5. 非终极结点上加了概率的树形图 T2

把结点上的相应规则的概率相乘，就可以计算出树形图 T2 的概率如下：

$$P(T2) = 0.8\times0.4\times0.05\times0.2\times0.4\times1.0\times1.0\times0.4\times0.05$$
$$= 2.56\times10^{-5}$$

树形图 T3 加了概率之后变为：

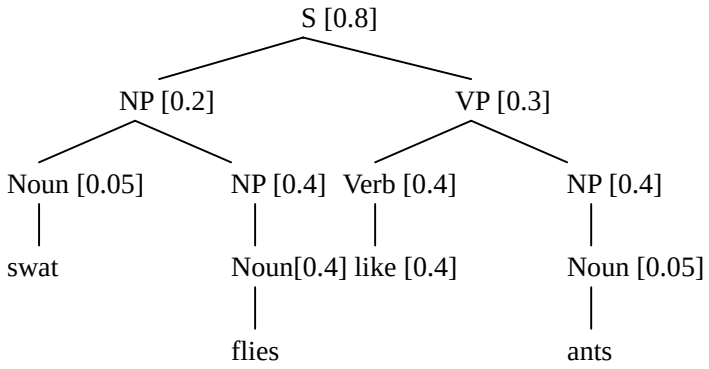


图 6. 非终极结点上加了概率的树形图 T3

把结点上的相应规则的概率相乘，就可以计算出树形图 T3 的概率如下：

$$P(T3) = 0.8\times0.2\times0.05\times0.4\times0.4\times0.3\times0.4\times0.4\times0.4\times0.05$$
$$= 1.2288\times10^{-6}$$

比较这三个树形图的概率，我们有：

$$P(T1) > P(T2) > P(T3)$$

根据树形图的概率，我们可以判定：“swat flies like ants”这个句子最可能的结构是树形图 T1 ,它的意思是：“像猛击蚂蚁一样地猛击苍蝇”。这个结论与我们的直觉是一致的，足见这个方法是可行的。因此，使用这样的方法，通过比较同一个有歧义的句子的不同树形图的概率，选择概率最大的树形图作为分析的结果，便可以达到歧义消解的目的。

这种歧义消解算法的实质是：从句子 S 的分析所得到的若干个树形图（我们把它们叫做 $\tau(S)$）中选出最好的树形图（我们把它叫做 T）作为正确的分析结果。

形式地说，如果 $T \in \tau(S)$,，那么，概率最大的树形图 T(S)将等于 argmax P(T)。我们有：

$$T(S) = \text{argmax } P(T)$$

计算出 argmax P(T)，就可以得到概率最大的树形图。可见，概率上下文无关语法是歧义消解的有力工具。

3. 概率 CYK 算法

在《基于短语结构语法的自动句法分析方法》(冯志伟，2000)中，我们曾介绍过 CYK 算法（Cocke-Younger-Kasami algorithm），CYK 算法是一种自底向上的动态规划剖析算法，经过概率上下文无关语法增强之后，就可以计算在剖析一个句子时的剖析概率。这样的算法叫做“概率 CYK 算法”。

首先，我们假定，概率上下文无关语法是具有 Chomsky 范式(Chomsky Normal Form)的，也就是说，如果一个语法中的每一个重写规则的形式或者为 A → BC，或者为 A → a，那么，这个语法就是具有 Chomsky 范式的语法。CYK 算法假定如下的输入、输出和数据结构：

- 输入。

  -- Chomsky 范式的 PCFG G = (N, Σ, P, S, D)。假定非终极符号|N|的索引号为 1, 2, ... |N|，初始符号的索引号为 1。

  -- n 个单词为 $w_1 \ldots w_n$。

- 数据结构。动态规划数组 π[i, j, a]表示跨在单词 i ... j 上的、非终极索引号为 a 的成分的最大概率。在这个区域上的反向指针用于存储剖析树中成分之间的链接。

● 输出。最大概率剖析将是 π [1, n, 1]：剖析树的根是 S，剖析树跨在单词 w_1 ... w_n 构成的整个符号串上。

与其他的动态规划算法一样，CYK 算法采用归纳法来填充概率数组，从基底出发，递归地进行归纳。为便于描写，我们用 w_{ij} 来表示从单词 i 到单词 j 的单词符号串。这样，我们有：

● 基底：我们考虑长度为 1 的输入符号串（也就是一个单词 w_i）。在 Chomsky 范式中，给定的非终极符号 A 展开为一个单词 w_i 的概率必定只是来自规则 A → w_i （因为当且仅当 A → w_i 是一个重写规则时，有 A ⇒ w_i）。

● 递归：对于长度大于 1（length > 1）的单词符号串，当且仅当至少存在一个规则 A → BC 以及某个 k，1≤k < j，使得 B 推导出 w_{ij} 的起头的 k 个符号串，C 推导出 w_{ij} 的后面 j －k 个符号串。因为这些符号串都比原来的符号串 w_{ij} 要短，它们的概率已经被存储在矩阵 π 中，我们把这两个片段的概率相乘，计算出 w_{ij} 的概率。当然，这时 w_{ij} 也可能会出现多个的剖析，所以，我们要在所有可能的剖析中（也就是在所有可能 k 的值和所有可能的规则中），选择概率最大的剖析作为我们的剖析结果。

仿照概率 CYK 算法，我们也可以做出概率 Earley 算法、概率线图分析法等。

4．规则和概率的获取

概率上下文无关语法的概率是从哪里来的？存在两种途径可以给语法指派概率。最简单的途径是使用句子已经得到剖析的语料库。这样的语料库叫做“树库”（tree-bank）。

如果我们已经加工并且建立了一个树库，语料库中的每一个句子都被剖析成相应的树形图，由于树形图中的每一个终极结点及其所管辖的字符串所构成的子树（sub-tree）相当于一条上下文无关语法中的重写规则，因此，我们可以对于树库中的所有树形图中所体现出来的这些上下文无关规则进行统计，就可以得出一部概率上下文无关语法。树库的质量越高，我们得到的概率上下文无关语法就越好。

例如，语言数据联盟（Linguistic Data Consortium）发布的宾州树库（Penn Treebank）（Marcus 等，1993），包括 Brown 语料库的剖析树，规模有 100 万单词，语料主要来自华尔街杂志（Wall Street Journal），部分语料来自 Switchboard 语料库。给定一个树库，一个非终极符号的每一个展开的概率都可以通过展开发生的次数来计算，然后将其归一化，就可以得到一部概率上下文无关语法。

但是，树库的加工和建立是非常困难的工作，随着语料库语言学的发展，更为可行的办法是通过未加工过的大规模语料库来自动地学习语法的规则，这样的自动学习，通常叫做“语法归纳”（grammar induction）。

对于一般的上下文无关语法，进行“语法归纳”时，自动学习的素材分为两部分，一部分是“正向训练实例”，一部分是“负向训练实例”。所谓正向训练实例，指的是语料库中那些真正属于该语言的句子或者其他类型的字符串。正向训练实例显然可以由一个语料库来提供。所谓负向训练实例指的是那些不属于该语言的字符串。人们在进行语法归纳时发现，如果不同时拥有正向训练实例和负向训练实例，那么，上下文无关语法的自动归纳就是不可能的。然而，目前我们还没有如何获取负向训练实例的有效手段，所以，对于一般的上下文无关语法，语法归纳是很困难的。

对于概率上下文无关语法，“语法归纳”问题实质上就是如何通过自动学习来获得一部带有概率的语法，从而使得正向训练实例中的句子的概率最大的问题，因此，不需要任何的负向训练实例就可以进行了。所以，在“语法归纳”时，概率上下文无关语法比一般的上下文无关语法更容易进行。

如果有一个未加工过的语料库，我们采用“向内向外算法”（inside-outside algorithm），

自动地从语料库中学习规则和概率，就可以得到一部概率上下文无关语法。在使用“向内向外算法”时，如果句子是没有歧义的，那么做法就很简单：只要剖析语料库就行了，在剖析语料库时，为每一个规则都增加一个计数器，然后进行归一化处理，就可以得到概率。但是，由于大多数句子都是有歧义的，在实际上我们必须为一个句子的每一个剖析都分别保持一个记数，并且根据剖析的概率给每一个局部的记数加权。向内向外算法是 Baker（1979）提出的，这种算法的完全描述，请参看 Manning 和 Schuetze（1999）。

一般的上下文无关语法的规则不考虑概率，规则一旦建立，就被认为是百分之百地成立的，是没有例外的，但是，由于语言具有创造性，即使用来自动学习的语料库再大，也难以保证获取的语法规则没有例外，语料库中总会有新的语法现象会超出已经确定的语法系统的规定。如果采用概率上下文无关语法，一个规则的成立往往不是百分之百的，它只在某个概率下成立，只要统计样本充分大，就可以保证概率有很高的准确性。对于那些在一般的上下文无关语法看来是例外的语言现象，概率上下文无关语法赋以它们比较小的概率，仍然承认他们存在的合理性。这样，概率上下文无关语法就可以合理地处理那些所谓“例外”的语言现象。

一般的上下文无关语法在识别句子时，只能给“合法”和“不合法”两种回答。合法的句子得到接受，不合法的句子遭到拒绝，非此即彼。这样的办法在分析真实语料时几乎寸步难行，因为在真实的语料中，很多句子的合法性是很难判定的，是亦此亦彼的，这种亦此亦彼的复杂情况往往使得自然语言处理系统处于进退两难的境地，不容易达到实用的要求。采用概率上下文无关语法，我们给合法的句子以较大的概率，给不合法的句子以较小的概率，这样，概率上下文无关语法就不仅能处理合法的句子，也能处理不合法的句子，它使语法摆脱了“非此即彼”的困境，给语法带来了“亦此亦彼”的柔性，使系统具备了容错的处理能力，而这样的容错处理能力对于实用的自然语言处理系统是非常重要的。

5. 概率上下文无关语法的无关性假设

为了能够使用加了概率的规则进行句法分析，概率上下文无关语法需要做如下的假设：**假设 1 -- 位置无关性假设**：子结点的概率与该子结点所直接管辖的字符串在句子中的位置无关。

为了便于说明，在非终极结点上加了概率的树形图 T1 中，我们给每一个非终极结点标上号码，得到如下的树形图：

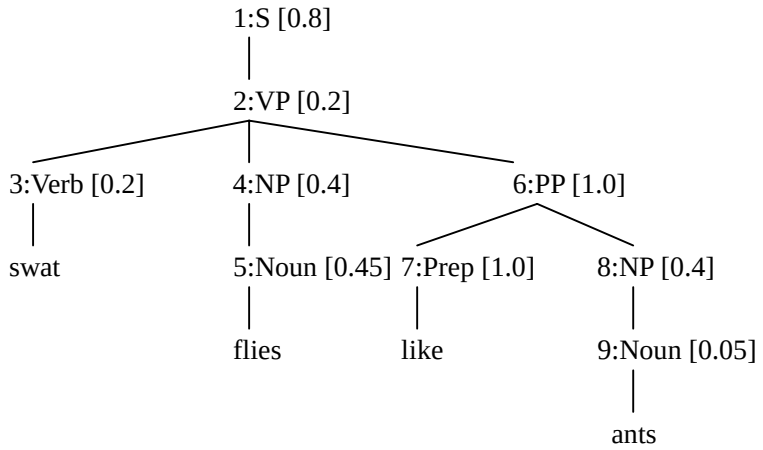


图 7. 结点上标了号码的树形图 T1

图 7 中，在这个树形图的位置 4，有一个规则 NP → Noun，在位置 8，也有一个规则 NP → Noun，尽管结点 NP 处在不同的位置，可是，由于这个结点 NP 直接管辖的字符串都是 Noun，所以，结点 NP 在这两个不同位置的概率都是相同的，都等于[0.4] 。也就是说，

结点的概率只与它所直接管辖的字符串 Noun 有关，而与 Noun 在句子中的位置无关。

假设 2 - 上下文无关性假设：子结点的概率与不受该子结点直接管辖的其他符号串无关。

例如，在图 7 的树形图中，如果把单词 swat 换成单词 kill，只会改变在位置 3 的结点 Verb 的概率，但是，不会改变这个树形图中不受位置 3 的结点 Verb 所直接管辖的其他结点的概率，也就是说，树形图中的其他结点 NP, PP 等的概率都保持不变。可见，单词的改变只对于直接支配该单词的非终极符号的概率有影响，而对于树形图中的其他非终极结点的概率没有影响。这个假设是上下文无关假设在概率方面的体现，它说明了，在概率上下文无关语法中，不仅重写规则是上下文无关的，而且，重写规则的概率也是上下文无关的。

假设 3 - 祖先结点无关性假设：子结点的概率与支配该结点的所有祖先结点的概率无关。

例如，在图 7 的树形图中，位置 4 的结点 NP 和位置 8 的结点 NP 的概率都是相同的，因为它们所直接管辖的字符串都是 Noun，可是，在位置 4 的结点 NP 的祖先结点是位置 2 的 VP 以及位置 1 的 S，在位置 8 的结点 NP 的祖先结点是位置 6 的 PP，这些祖先结点的概率都不会影响在位置 4 和在位置 8 的结点 NP 的概率。

由于有这三个假设，概率上下文无关语法就不仅继承了一般的上下文无关语法的上下文无关的特性，还使得概率值也具备了上下文无关的特性，这样，我们就可以利用概率上下文无关语法进行句法剖析（parsing）。首先使用通常的上下文无关语法的分析算法来剖析句子，得到句子的句法剖析树形图；然后，给每一个非终极结点加上一个概率值，在上述三个假设下，每一个非终极结点的概率值也就是对该非终极结点进一步重写所使用的规则后面附带的概率，我们得到的树形图是带有概率的树形图。如果句子是有歧义的，我们就会得到不同的带有概率的树形图，比较这些树形图的概率，选择概率最大的树形图作为句法剖析的结果，就可以达到对句子进行歧义消解的目的。

6. 结构依存和词汇依存的问题

然而，概率上下文无关语法并不是完美无缺的，它还存在结构依存和词汇依存的问题。

6.1 结构依存问题

根据上述的三个无关性假设，在概率上下文无关语法中，对规则左部的非终极符号进行重写时，不依赖于其他的非终极符号。正是由于在概率上下文无关语法中，每一条规则都是独立的，所以规则的概率才可以相乘。

然而，在英语中，结点上规则的转写与结点在树形图中的位置是有关的。例如，英语句子中的主语倾向于使用代词，这是因为主语通常是表示主题或者旧信息，而要援引旧信息时往往使用代词，而不是代词的其他名词往往用于引入新信息。根据 Francis (1999)的调查，在 Switchboard 语料库中，陈述句的主语有 31,021 个，其中 91%的主语是代词，只有 9%的主语是其他词。与此相反，在 7,498 个宾语中，只有 34%是代词，而 66%是其他词。

主语：： **She** is able to take her baby to work with her. [代词做主语，占 91%]

My wife worked until we had a family. [非代词做主语，只占 9%]

大部分的主语是代词。

宾语： Some laws absolutely prohibit **it**. [代词做宾语，占 34%]

All the people signed **applications**. [非代词做宾语，占 66%]

大部分的宾语是非代词。

这样的语言事实是对概率上下文无关语法的上述无关性假设的严重挑战。根据无关性假设，概率上下文无关语法不能处理这样的语言现象。

6.2 词汇依存问题

⑴ PP 附着：在英语句子中，介词短语 PP 可以做中心动词短语 VP 的状语，也可以做它前面名词短语 NP 的修饰语，究竟是附着于 VP，还是附着于 NP，这就是所谓“PP-附着”

（PP-attachment）问题。PP-附着与词汇有关。

例如，在句子 “Washington sent more than 10,000 soldiers into Afghanistan”中, 介词短语（PP） “into Afghanistan”或者附着于名词短语（NP） “more than 10,000 soldiers”，或者附着于动词短语（VP）“sent”(单独的动词也可以看成一个动词短语)。这里存在 PP-附着问题。

在概率上下文无关语法中，这种 PP-附着的判定要在下面的规则之间进行选择：

NP → NP PP　　(PP 附着于 NP)

和　　VP → VP PP　　(PP 附着于 VP)

这两个规则的概率依赖于训练语料库。在训练语料库中，NP 附着和 VP 附着的统计结果如下：

语料库	PP 附着于 NP	PP 附着于 VP
AP Newswire (13 00 万词)	67%	33%
Wall Street Journal & IBM manuals	52%	48%

可以看出，在两个训练语料库中，“PP 附着于 NP”都处于优先地位。根据这样的统计结果，我们应该选择 PP 附着于 NP，也就是选择 PP “into Afghanistan”附着于 NP “more than 10,000 soldiers”这个结果。但是，在我们上面的句子中，介词短语“into Afghanistan”的正确附着却应该是附着于动词短语 VP(“sent”)，这是因为这个 VP“sent”往往要求一个表示方向的介词短语 PP，而介词短语“into Afghanistan”正好满足了这个要求。概率上下文无关语法显然不能处理这样的词汇依存问题。

⑵　并列结构的歧义：

句子“dogs in houses and cats”是有结构歧义的：

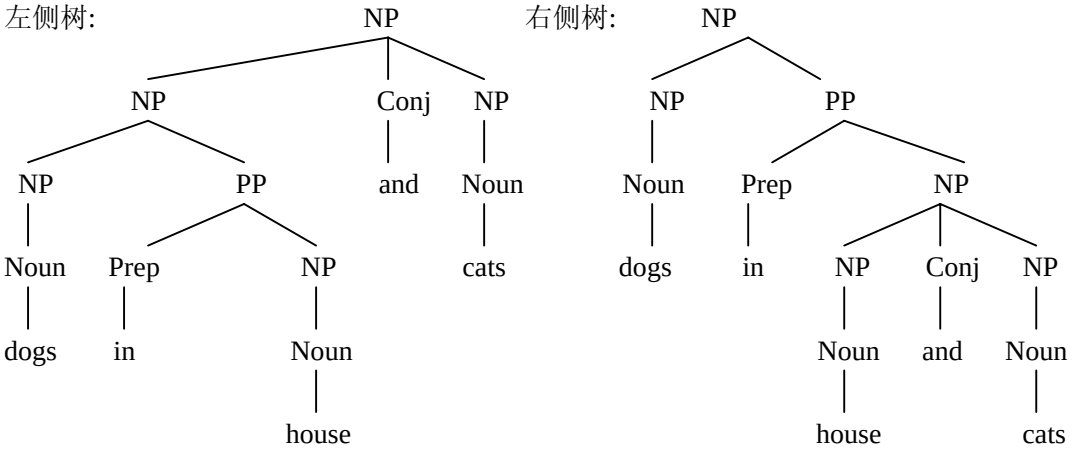


图 8 并列结构歧义

尽管在直觉上上我们认为左侧树是正确的，但是，由于左右两侧的树所使用的规则是完全一样的，这些规则如下：

NP → NP Conj NP

NP → NP PP

NP → Noun

PP → Prep NP

Noun → dogs

Noun →house

Noun → cats

Prep → in

Conj → and

根据上述的无关性假设，由于规则完全相同，使用这些规则的概率相乘而计算出来的两个树形图的概率也应该是一样的。在这种情况下，概率上下文无关语法将指派这两个树形图以相同的概率，也就是说，概率上下文无关语法无法判定这个句子的歧义。

由此可见，概率上下文无关语法在遇到结构依存和词汇依存问题的时候就显得捉襟见肘、无能为力了，我们还需要探索其他的途径来进一步提升概率上下文无关语法的功能，其中的一个有效的途径，就是在概率上下文无关语法中引入词汇信息，采用词汇中心语概率表示法，把概率上下文无关语法提升为概率词汇化上下文无关语法。

7. 概率词汇化上下文无关语法

Charniak (1997) 提出了词汇中心语概率表示的方法。他的方法实际上是一种词汇语法（lexical grammar.），这种语法也叫做概率词汇化上下文无关语法(Probabilistic Lexicalized Context-Free Grammar)。在 Charniak 的概率表示中，剖析树的每一个结点要标上该结点的中心词（head）。例如，句子“Workers dumped sacks into a bin” 可表示如下：

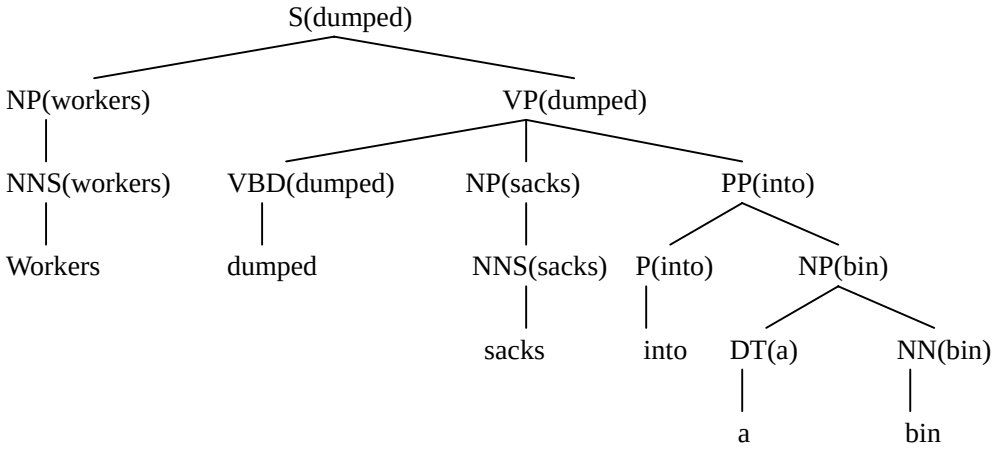
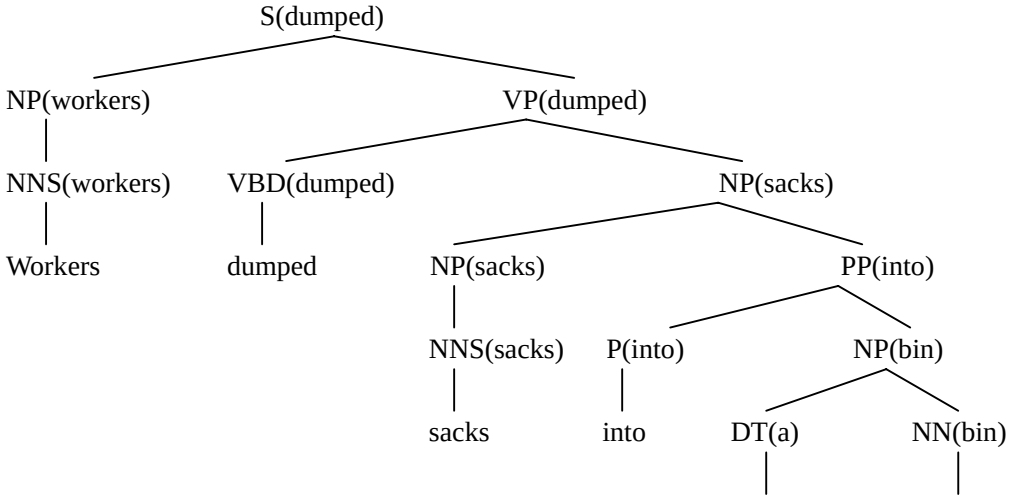


图 9 词汇化的剖析树

这时，概率词汇化上下文无关语法的规则数目将比概率上下文无关语法的规则多得多。例如，我们可以有如下的规则，规则中既包括概率，也包括词汇信息：

VP(dumped) → VBD(dumped) NP(sacks) PP(into) [3×10^{-10}]
VP(dumped) → VBD(dumped) NP(cats) PP(into) [8×10^{-11}]
VP(dumped) → VBD(dumped) NP(hats) PP(into) [4×10^{-10}]
VP(dumped) → VBD(dumped) NP(sacks) PP(above) [1×10^{-12}]

这个句子也可以被剖析为另一个树形图，不过，这个树形图是不正确的：



a bin

图 10 不正确的剖析树

如果我们把 VP(dumped) 重写为 VBD NP PP，那么，我们可以得到正确的剖析树。如果我们把 VP(dumped) 重写为 VBD NP，那么，就得到上面的这个不正确的剖析树。

我们可以根据 Penn Tree-bank 中的 Brown 语料库来计算这种词汇化规则的概率：

第一个词汇化规则 VP(dumped)→VBD NP PP 的概率为：

$$P(VP \rightarrow VBD\ NP\ PP | VP, dumped) = \frac{C(VP(dumped) \rightarrow VBD\ NP\ PP)}{\sum_{\beta} C(VP(dumped) \rightarrow \beta)}$$

$$= 6/9 = .67$$

第二个词汇化规则 VP(dumped)→VBD NP 从不在 Brown 语料库中出现，因为“dump”这个动词要求指明动作所到达的新的位置，因此，如果它后面没有介词短语，就是不合理的。

$$P(VP \rightarrow VBD\ NP | VP, dumped) = \frac{C(VP(dumped) \rightarrow VBD\ NP)}{\sum_{\beta} C(VP(dumped) \rightarrow \beta)}$$

$$= 0/9 = 0.$$

在实际的应用中，如果概率出现零值，一般都要进行平滑（smoothing），为简单起见，这里我们不考虑平滑问题。

由于第二个词汇化规则的概率为零，所以，使用这个规则得到的图 10 中的剖析树是不正确的。

我们也可以用同样的方法来计算中心词的概率。

在正确的剖析树中，结点 PP 的母亲结点(X)是中心词“dumped”，在不正确的剖析树中，结点 PP 的母亲结点(X)是中心词“sacks”

根据 Penn Tree-bank.的 Brown 语料库，我们有

$$P(into | PP, dumped) = \frac{C(X(dumped) \rightarrow \ldots PP\ (into) \ldots)}{\sum_{\beta} C(X(dumped) \rightarrow \ldots PP \ldots)}$$

$$= 2/9 = .22$$

$$P(into | PP, sacks) = \frac{C(X(sacks) \rightarrow \ldots PP\ (into) \ldots)}{\sum_{\beta} C(X(sacks) \rightarrow \ldots PP \ldots)}$$

$$= 0/0 = ?$$

可见，通过计算 PP 结点的母亲结点的概率，也可以判断 PP(into)修饰 dumped 的概率比修饰 sacks 的概率大。

当然，只是一个例子还不能证明一个方法一定比其他的方法好。另外，我们上面提到的概率词汇语法只是 Charniak 的实际算法的一个简化版本。他还增加了一些附加的条件因素（例如，某结点的祖父结点句法范畴的规则展开概率），并提出了各种回退与平滑算法，不过，现有的语料库要获取这些统计数字还是显得太小了。另外一些统计剖析器包括更多的因素，例如，区分论元成分（argument）与附属成分（adjunct），对于树形图中那些比较接近的词汇依存关系比那些比较疏远的词汇依存关系给以更大的权重（Collins, 1999），考虑在给定成分中的三个最左的词类（Magermen 和 Marcus, 1991），以及考虑一般的结构依存关系（例如，英语中右分支结构优先）（Briscoe 和 Carroll, 1993），等等。这些方法都比较专门，限于篇幅，这里就不再赘述了。

概率上下文无关语法和概率词汇化上下文无关语法对于规则方法和统计方法的结合，进

行了有成效的探索，大大地增强了上下文无关语法消解歧义的能力。这样的概率语法是当代计算语言学的一个新成果，应该引起我们的注意。

参考文献

Booth, T. L., 1969, Probabilistic representation of formal languages, IN IEEE Conference Record of the 1969 Tenth Annual Symposium on Switching and Automata Theory.

Briscoe, T., and Carroll, J., 1993, Generalized probabilistic LR parsing of natural language (corpora) with unification-based grammar, Computational Linguistics, 19 (1).

Charniak, E, 1993, Statistic Language Learning, MIT Press.

———— , 1997, Statistic parsing with a context-free grammar and word statistics, In AAAI-97, Menlo Park, AAAI Press.

Collins, M. J., 1999, Head-driven Statistic Models for Natural Language Processing, Ph.D thesis, University of Pennsylvania University, Philadelphia.

Francis, H.S., Gregory, M.L., and Michaelis, L.A., 1999, Are lexical subject deviant? In CLS-99, University of Chicago.

Magerman, D. M., and Marcus, M. P., 1991, Pearls: A probabilistic chart parser, In proceedings of the 6th Conference of the European Chapter of the ACL, Berlin, Germany.

Manning, C. D., and Schuetze, H., 1999, Foundation of Statistic Natural Language Processing, MIT Press, Cambridge MA.

冯志伟，2000，基于短语结构语法的自动句法分析方法，《当代语言学》，第 2 期。

———，2002，线图分析法，《当代语言学》，第 4 期。

———，2003a，一种无回溯的自然语言分析算法，《语言文字应用》，第 1 期。

———，2003b，花园幽径句的自动分析算法，《当代语言学》，第 4 期。