

第三讲 附录

当代形式语法理论概观

詹卫东

<http://ccl.pku.edu.cn/doubtfire/>

当代形式语法理论概观

- TG, GB, MP,
- LFG, GPSG, HPSG,
- PATR-II, DCG, FUG,
- 树邻接语法 (TAG)
- 链语法 (Link Grammar)
- 范畴语法 (Categorial Grammar)
- 依存语法 (Dependency Grammar)
- 词语法 (Word Grammar)
-

当代形式语法理论的不同阵营

当代形式语法理论体系的分类

分类角度：

(1) 从目标来看：

理论型语法系统

应用型语法系统

做了语法模型
干什么用呢？

(2) 从手段来看：

产生式规则 + 特征结构

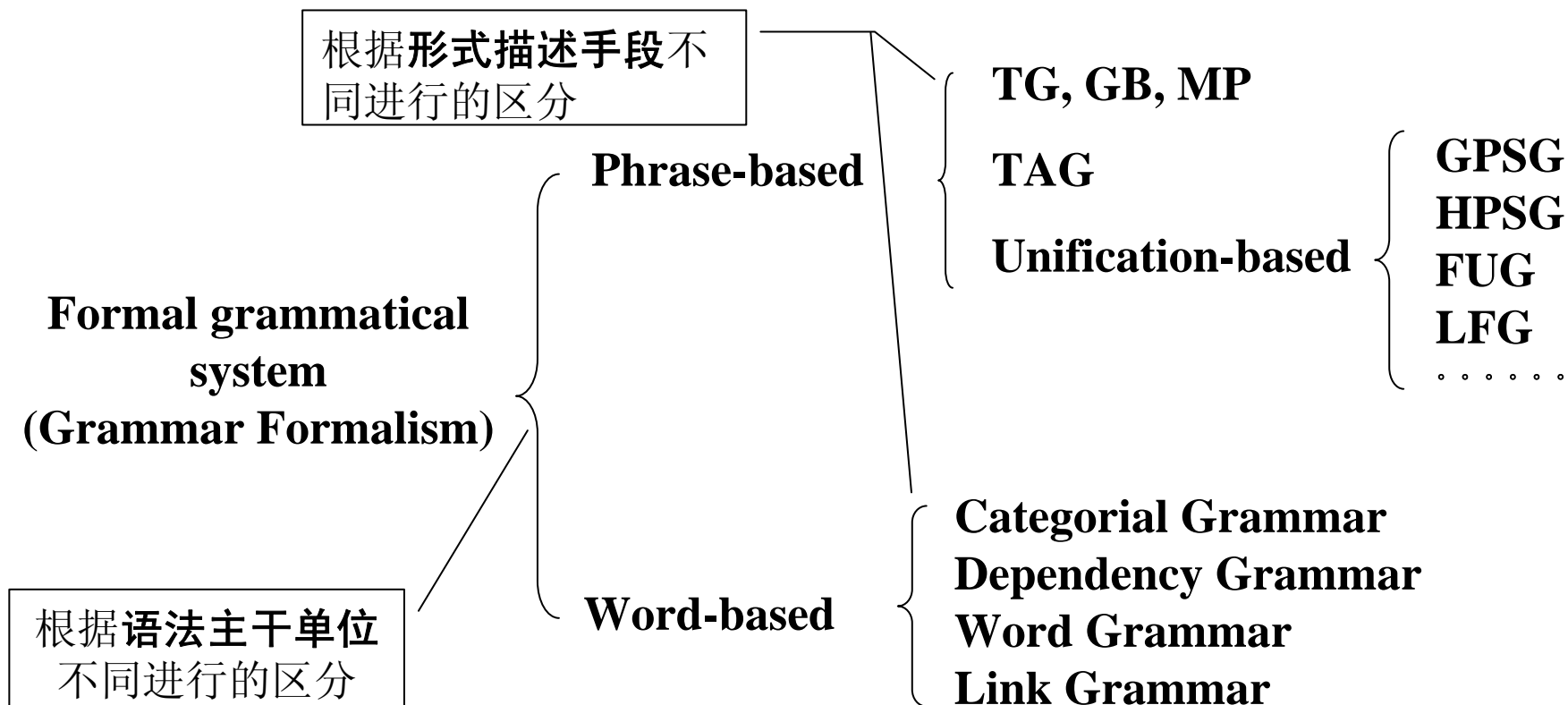
树结构

语法知识的词汇化

自动机

剪刀! 锤子! 布! 这
么多工具, 用哪个
建模型呢?

一个根据语法的主干单位不同和形式描述手段不同得出的分类层级



理论型语法系统

采用转换规则

TG(1957), GB(1981), MP(1992)
— Chomsky派

1假设深层结构;
2仅用原子范畴

LFG(1982)
GPSG(1985)
HPSG(1987) } — 非Chomsky派

1不假设深层结构;
2广泛采用复杂特征

不引入转换规则

应用型语法系统

- **PATR-II** (Stuart M. Shieber, 1984, 1986)
CFG + [特征结构 / 合一式 / 函数] (相互独立)
- **DCG** (F.C.N. Pereira & D.H.D. Warren, 1980)
语法规则可以转换为Prolog程序语言
- **FUG** (Martin Kay, 1979, 1985)
基于FD的规则 + 条件测试 (融合)
- **Lingware** (德国Sail Labs公司语言知识规则描述语言)
Lisp程序语言风格的规则

1 LFG（词汇功能语法）

- ◆ Lexical Function Grammar
- ◆ Kaplan, R. and J.Bresnan, 1982, *Lexical-Functional Grammar : A Formal System for Grammatical Representation*, In J.Bresnan ed. *The Mental Representation of Grammatical Relations*, MIT Press 1982.
- ◆ <http://www-lfg.stanford.edu/lfg/bresnan/>
- ◆ <http://www.parc.xerox.com/istl/members/kaplan/>

基本特点

- ◆ 基本思想：依托短语结构语法已有的树结构，通过自底向上（**bottom-up**）层层传递的方式把词汇所负载的各种信息传播、汇集到上层节点中去，最终形成关于一个句子的完整的结构信息和功能信息描述。
- ◆ 分析结果：
 - C-结构（constituent structure）：对句子的句法结构树描述
 - F-结构（functional structure）：对句中各成分的功能描述

带“修饰”的Rewriting Rule

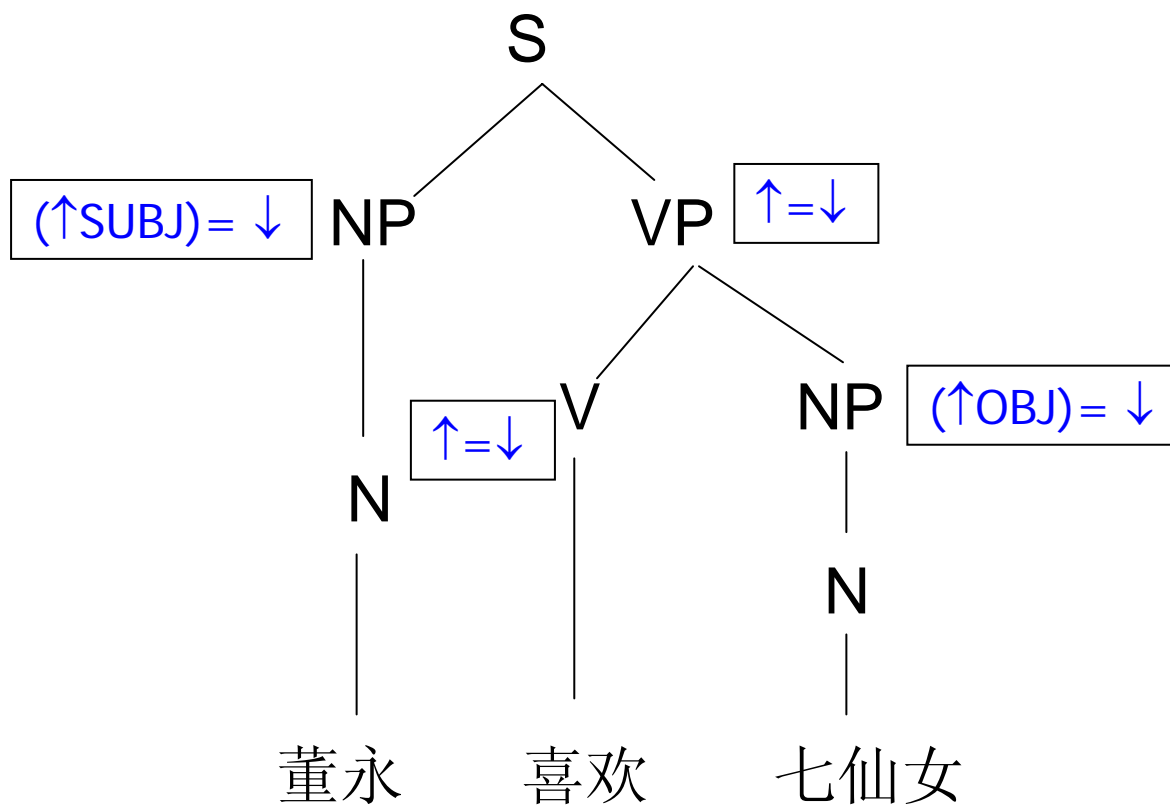
S →	NP	VP
	(↑SUBJ)= ↓	↑=↓
VP →	V	NP
	↑=↓	(↑OBJ)= ↓
VP →	V	
	↑=↓	
NP →	N	
	↑=↓	

↑ 表示父
亲节点；
↓ 表示当
前节点；
↑SUBJ 表
示父亲节
点的SUBJ
特征； =
表示合一

词汇信息描述（特征结构）

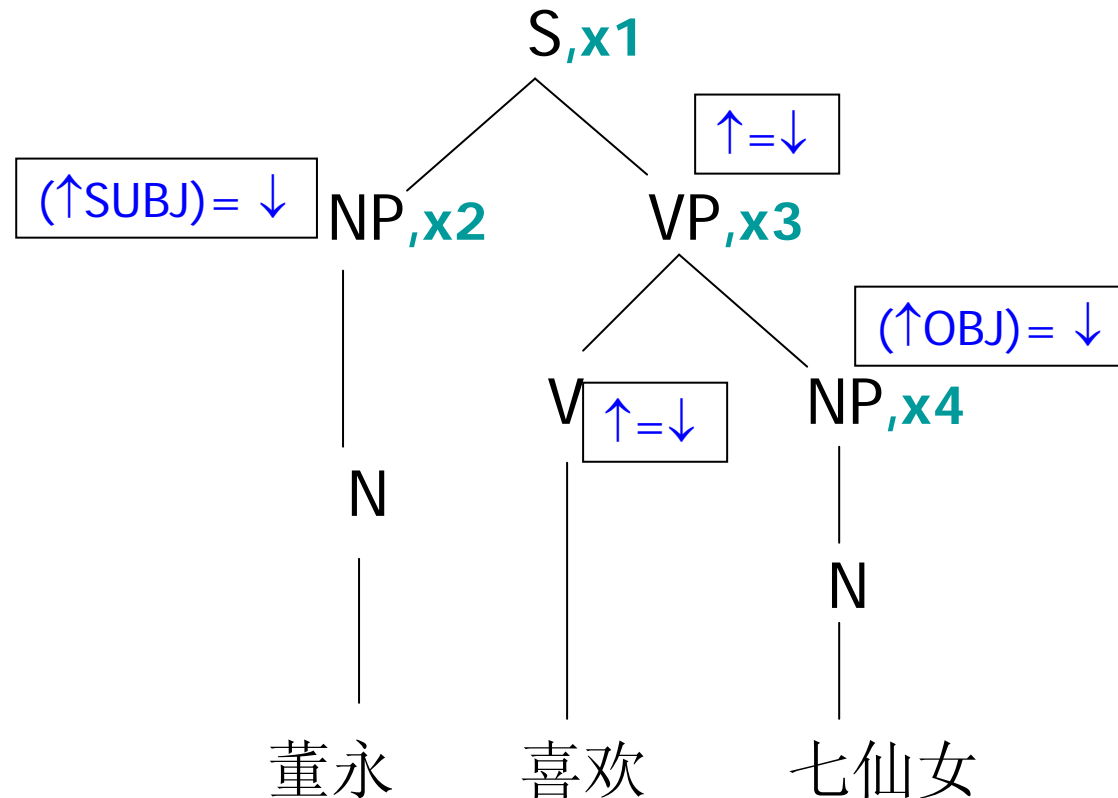
董永	N	(↑LEX)='董永' (↑SEM)=HUMAN (↑PERS)=3
七仙女	N	同上
喜欢	V	(↑PRED)='喜欢<(↑SUBJ), (↑OBJ)>' (↑SUBJ SEM)=HUMAN (↑OBJ SEM)=HUMAN

C-结构



从C—结构到F—结构

- 从成分结构导出功能方程；
- 从功能方程求解功能结构；



从成分结构导出功能方程

1. $x1.SUBJ = x2$
2. $x1 = x3$
3. $x3.OBJ = x4$
4. $x2.LEX = \text{“董永”}$
5. $x2.SEM = HUMAN$
6. $x2.PERS = 3$
7. $x3.PRED = \text{“喜欢<(x3 SUBJ),(x3 OBJ)>”}$
8. $x3.SUBJ.SEM = HUMAN$
9. $x3.OBJ.SEM = HUMAN$
10. $x4.LEX = \text{“七仙女”}$
11. $x4.SEM = HUMAN$
12. $x4.PERS = 3$

. 相当于“的”

从功能方程求解功能结构

$$x_1 = x_3 = \left[\begin{array}{l} PRED = \text{喜欢} < SUBJ, OBJ > \\ SUBJ = x_2 \\ OBJ = x_4 \end{array} \right]$$

$$x_2 = \left[\begin{array}{l} LEX = \text{董永} \\ SEM = HUMAN \\ PERS = 3 \end{array} \right]$$

$$x_4 = \left[\begin{array}{l} LEX = \text{七仙女} \\ SEM = HUMAN \\ PERS = 3 \end{array} \right]$$

“董永喜欢七仙女”的功能结构

$$FS(x_1) = \left[\begin{array}{l} PRED = \text{喜欢} \langle SUBJ, OBJ \rangle \\ \\ SUBJ = \left[\begin{array}{l} LEX = \text{董永} \\ SEM = HUMAN \\ PERS = 3 \end{array} \right] \\ \\ OBJ = \left[\begin{array}{l} LEX = \text{七仙女} \\ SEM = HUMAN \\ PERS = 3 \end{array} \right] \end{array} \right]$$

LFG检查句子合法性的标准

- 一个属性只允许有一个值（唯一性）；
- 每个属性都应该有值（完备性）；
- 不该有的属性不应该有值（一致性）；

2 GPSG（广义短语结构语法）

- Generalized Phrase Structure Grammar
- Gerald Gazdar, Ewan Klein, Geoffrey K. Pullum, Ivan A. Sag, 1985, *Generalized Phrase Structure Grammar*, Oxford, England, Blackwell Publishing and Cambridge, 1985.
- <http://www.cogs.susx.ac.uk/lab/nlp/gazdar/gazdar.html>

基本特点

- 语法模型：
基本规则 -> 合格性检查 -> 树结构
- 基本规则：
 - 直接支配规则（Immediate Dominance Rule）
 - 元规则（Meta Rule）
- 合格性检查：
 - 特征共现限制（Feature Co-occurrence Restriction）
 - 线性顺序描述（Linear Precedence Statements）
 -

3 HPSG (中心词驱动的短语结构语法)

- Head-Driven Phrase Structure Grammar
- Pollard, Carl and Ivag A. Sag. 1987. *Information Based Syntax and Semantics*. CSLI Lecture Notes, No.13, The University of Chicago Press, Chicago.
- Pollard, Carl and Ivag A. Sag. 1994. *Head-Driven Phrase Structure Grammar*. The University of Chicago Press, Chicago.
- Sag, Ivan A. & Thomas Wasow, 1999, ***Syntactic Theory: A Formal Introduction***, CSLI Publications, Stanford, California.
- <http://ling.ohio-state.edu/research/hpsg/>
- <http://hpsg.stanford.edu/>

基本特点

- 强调中心词在短语结构规则中的作用
 - 中心语 — 补足语规则 (Head - Complement Rule)
 - 中心语 — 指示语规则 (Head - Specifier Rule)
 - 中心语 — 修饰语规则 (Head - Modifier Rule)
- 产生式规则 + 特征结构 + 合一运算
- 基于中心词的属性特征传递 (Head Feature Principle, ...)
- 以同样的形式化方式表达句法知识和语义知识

4 PATR-II

- Shieber, Stuart M., 1984, *The Design of a Computer Language for Linguistic Information*, In Proceedings of Coling84(10th), Stanford University, Stanford, California.
- Shieber, Stuart M., 1985, *Using Restriction to Extend Parsing Algorithms for Complex-feature-based Formalisms*, In Proceedings of the 22nd Annual Meeting of the ACL, University of Chicago, Chicago, Illinois.
- Shieber, Stuart M., 1986, *An Introduction to Unification Based Approaches to Grammar*, CSLI Lecture Notes Series, No.4, Stanford University.
- <http://www.eecs.harvard.edu/~shieber/>

基本特点

- 在独立的**CFG**基础上增加特征结构与合一运算
- 在规则中引入附加范畴（比如通配符 `*. %`）
- 在合一式中引入函数（比如 `morph_gen()`）
- 引入逻辑表达符号增强规则表达能力（比如多个合一式之间可以有析取关系）

5 DCG (定子句语法)

- Definite Clause Grammar
- Pereira, F.C.N., and D.H.D. Warren, 1980, *Definite Clause Grammars for Language Analysis - A Survey of the Formalism and a Comparison with Augmented Transition Networks*, In *Artificial Intelligence*, Vol.13, pp231-278, 1980.

基本特点

标准DCG中为 -->

- DCG语法可以方便地翻译为Prolog程序语言

C
F
G

S -> NP VP
VP-> V NP
NP-> N
N->董永

s(P1,P3) :- np(P1,P2), vp(P2,P3).
vp(P1,P3) :- v(P1,P2), np(P2,P3).
np(P1,P2) :- n(P1,P2).
is_n(董永).

D
C
G

₁董永₂喜欢₃七仙女₄

→

word(董永,1,2).
word(喜欢,2,3).
word(七仙女,3,4).

分析“董永喜欢七仙女”

`n(From,To) :- word(Word,From,To), is_n(Word).`

`v(From,To) :- word(Word,From,To), is_v(Word).`

`?- s(1,4).`

`** Yes.`

增加一条规则，描述短语内部结构

`s(P1,P3,s(NP,VP)) :- np(P1,P2,NP), vp(P2,P3,VP).`

`s(1, 4, s(np(n(董永)),vp(v(喜欢), np(n(七仙女))))).`

得到短语内部结构的分析结果

6 FUG (功能合一语法)

- Functional Unification Grammar
- Martin Kay, 1979, ***Functional Grammar***, In Proceedings of the 4th Annual Meeting of the Berkeley Linguistics Society.
- Martin Kay, 1985, *Parsing in Functional Grammar*, In D.Dowty, L. Karttunen, and A. Zwicky eds, Natural Language Parsing, Cambridge University Press, Cambridge, 1985.

基本特点

- 弱化线性序结构关系
- 强调功能结构
- 对所有语言单位统一采用**FD**形式描述
- 适合于生成（**Generation**）

7 TAG (树邻接语法)

- A. Joshi, L. Levy, & M. Takahashi, 1975, **Tree Adjunct Grammar**, Journal of Computer & System Science, 1975, 10(1): pp136-163.
- Anne Abeillé, Owen Rambow, 2000, Tree Adjoining Grammars : Formalisms, Linguistic Analysis and Processing, CSLI Publications.
- <http://www.cis.upenn.edu/~xtag/>

TAG与一般CFG的区别

TAG跟一般CFG的区别在于：针对自然语言的知识表示需要，将树结构写得更细，比如：

$P_{CFG}: VP \rightarrow VP NP \quad (R1)$

$VP \rightarrow V NP \quad (R2)$

$P_{TAG}: VP \rightarrow VP(V NP) NP \quad (R)$

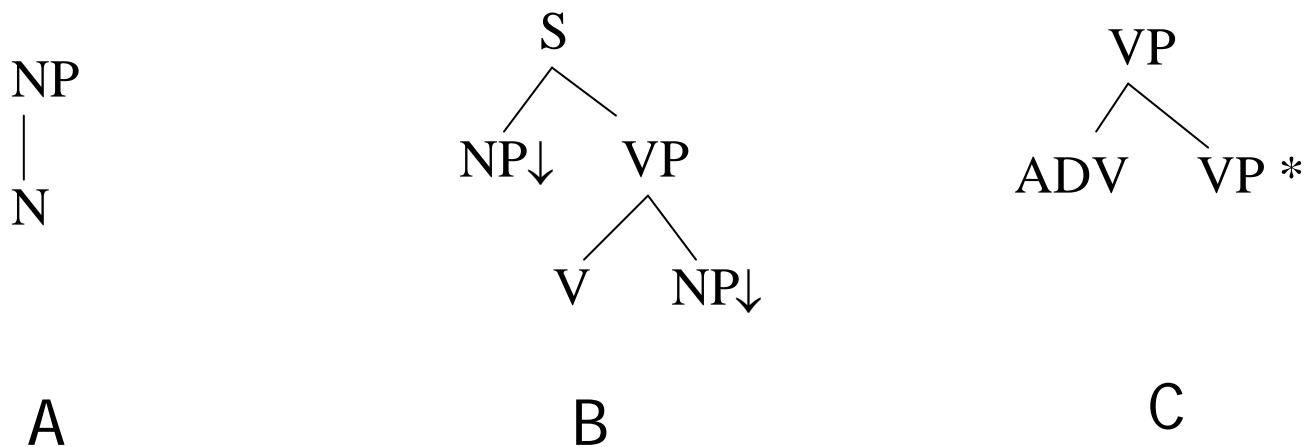
so called “mildly context sensitive grammar formalism”

为了描写自然语言中的双宾结构，CFG与TAG采取不同的规则

基本概念

- 基本语法单位：基本树结构 (Elementary Tree)
 - 初始树 (Initial Tree)
 - 辅助树 (Auxiliary Tree)
- 树上的基本操作
 - 替换 (Substitution)
 - 插接 (Adjunction)

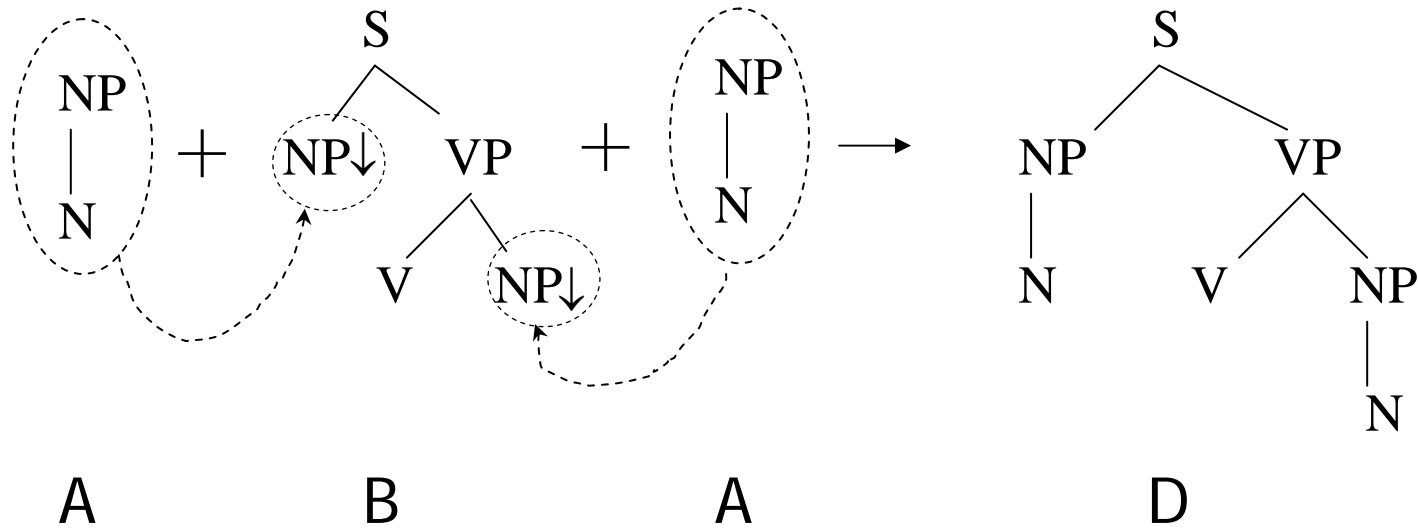
初始树 与 辅助树



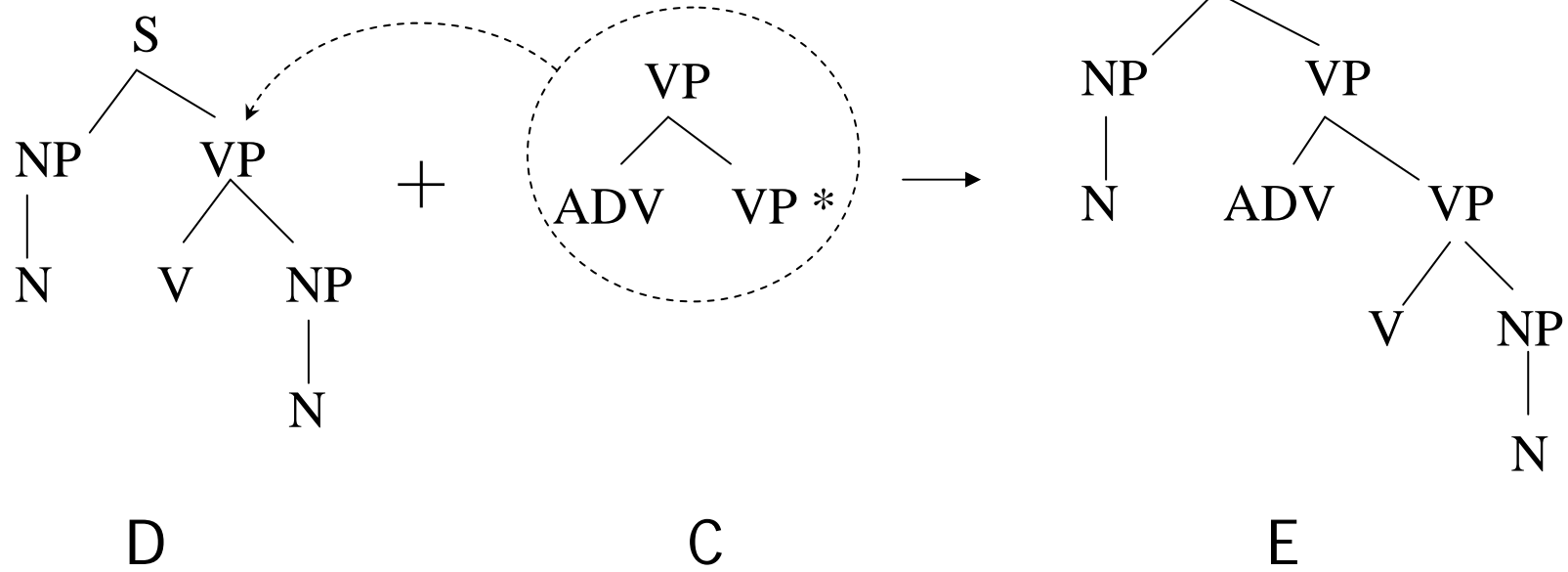
↓ 表示该节点可被替换

* 表示该节点所在的树可以插接到另一棵树的同名节点位置

替换



插接



TAG与CFG比较

I: $\alpha_1 =$
S
|
e

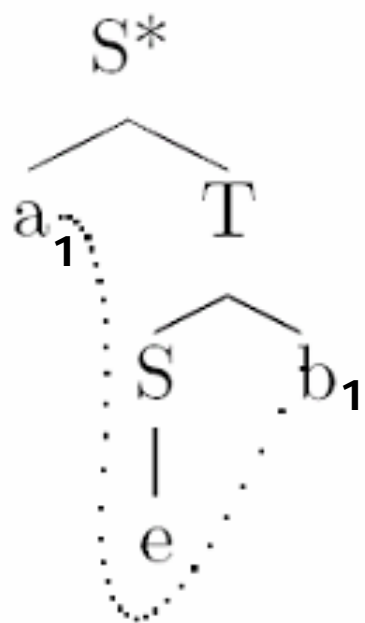
A: $\beta_1 =$
S
/ \
a T
 / \
 S b

$\beta_2 =$
T
/ \
a S
 / \
 T b

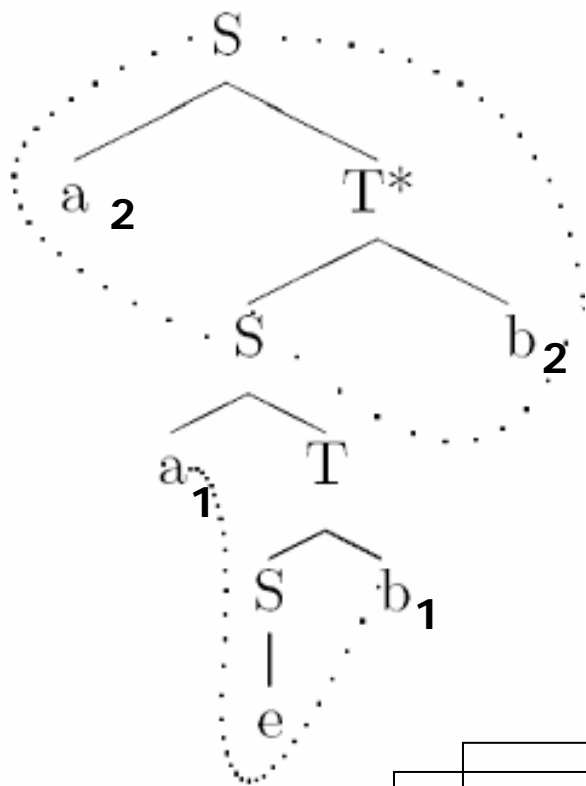
I: $\alpha_1 =$
 $\begin{array}{c} S \\ | \\ e \end{array}$

A: $\beta_1 =$
 $\begin{array}{c} S \\ / \quad \backslash \\ a \quad T \\ \quad \quad / \quad \backslash \\ \quad \quad S \quad b \end{array}$

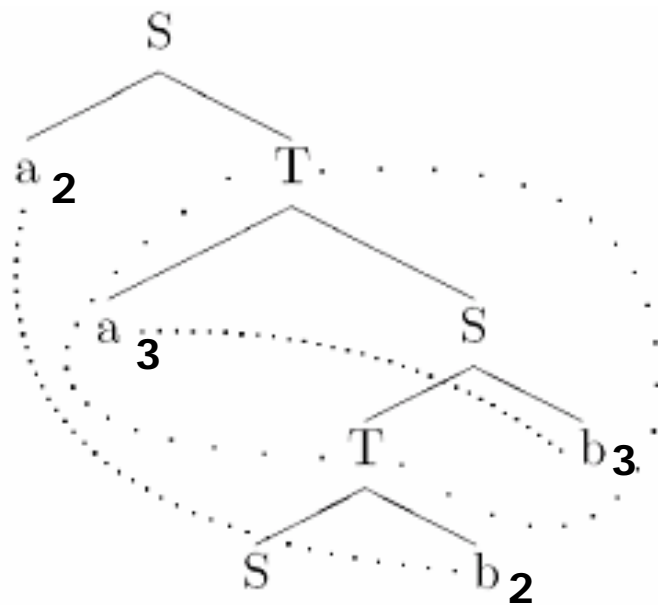
$\beta_2 =$
 $\begin{array}{c} T \\ / \quad \backslash \\ a \quad S \\ \quad \quad / \quad \backslash \\ \quad \quad T \quad b \end{array}$



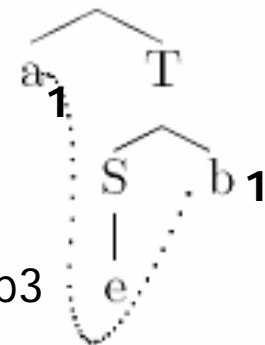
a1 e b1



a2 a1 e b1 b2



a2 a3 a1 e b1 b2 b3



TAG 规则

T1) $S \rightarrow e$

T2) $S \rightarrow a T(S b)$

T3) $T \rightarrow a S(T b)$

CFG 规则

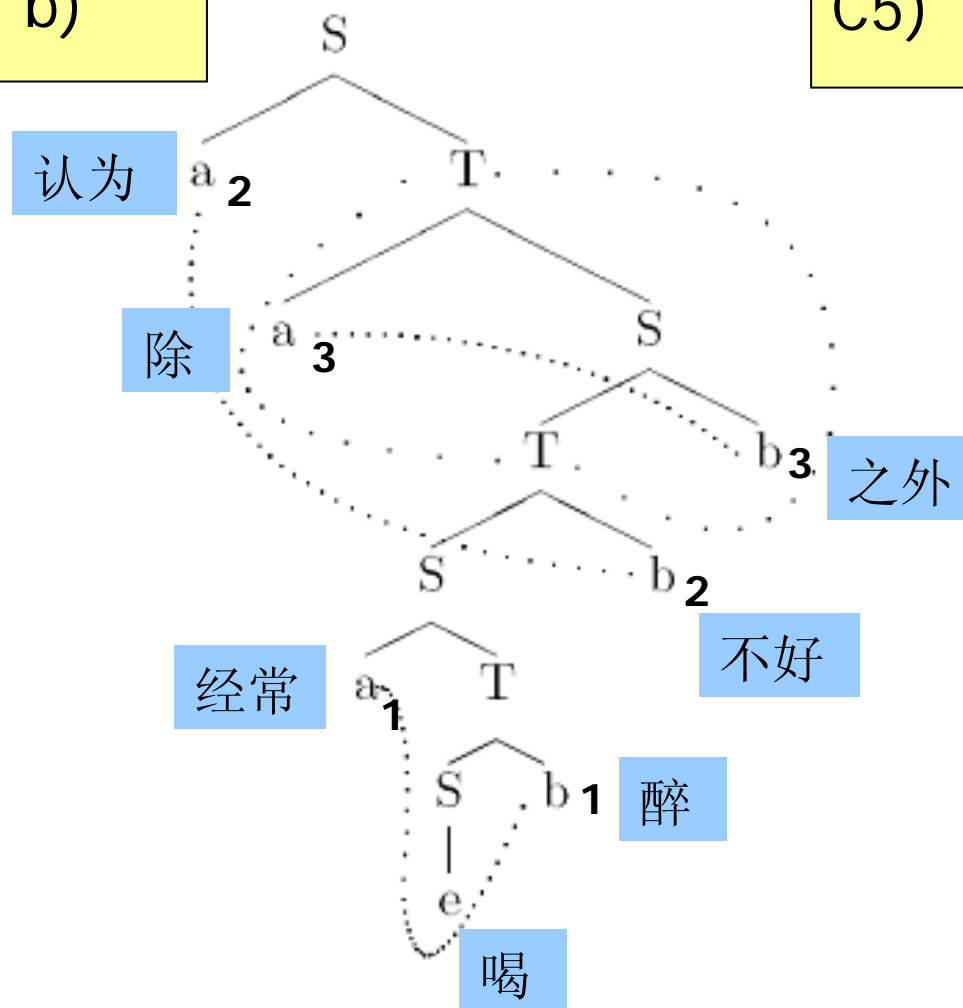
C1) $S \rightarrow e$

C2) $S \rightarrow a T$

C3) $S \rightarrow T b$

C4) $T \rightarrow S b$

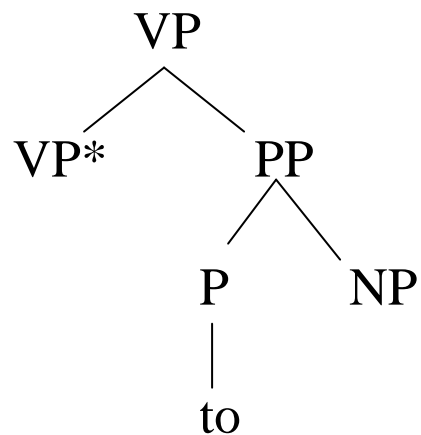
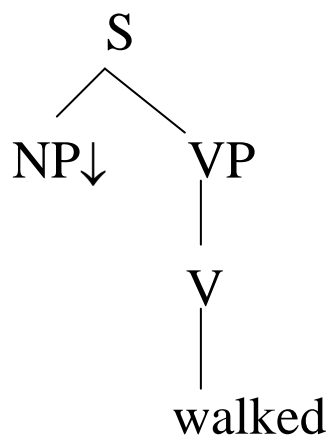
C5) $T \rightarrow a S$



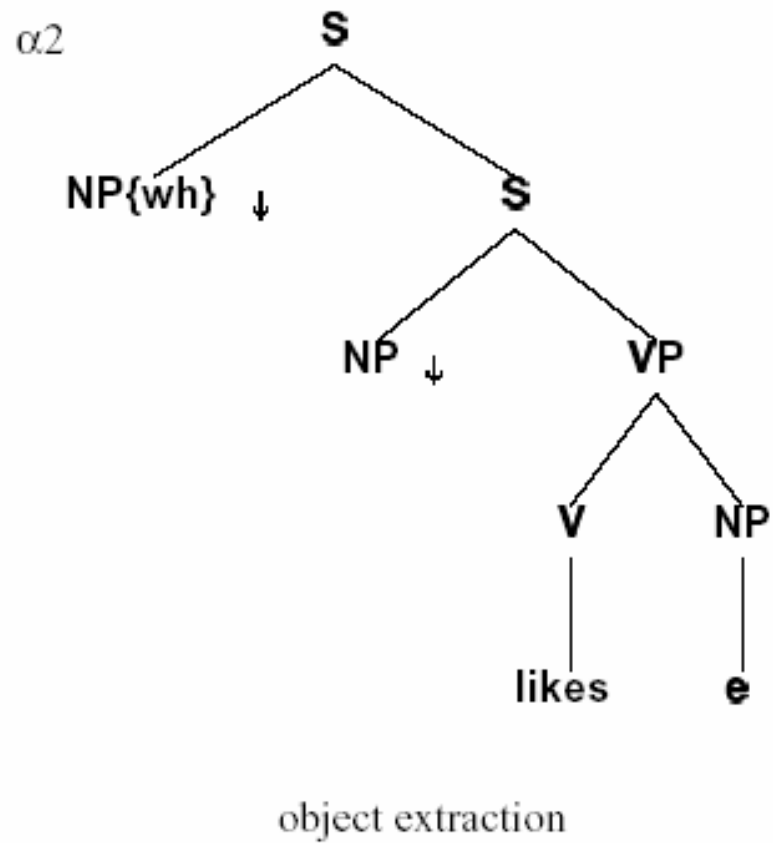
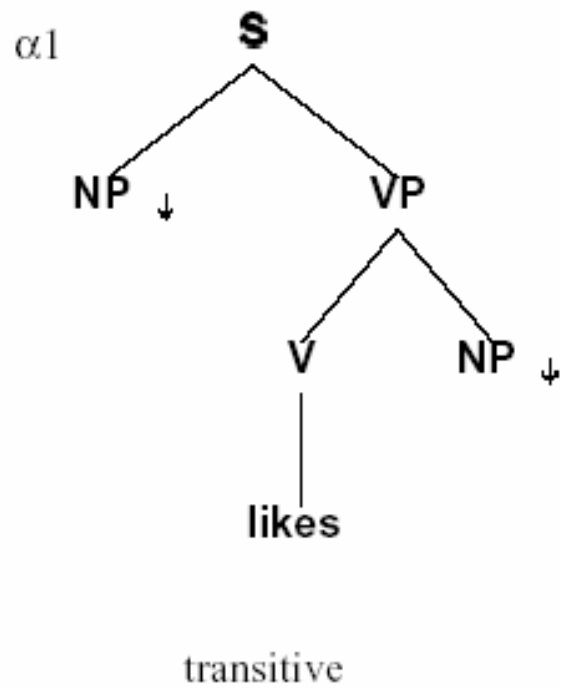
LTAG (Lexicalized TAG)

- **LTAG**对**TAG**的扩展主要是将每个初始树和辅助树都跟某个和某一些具体的词相关联起来。
- 一个**LTAG**树结构中的“词”节点被称作是这个树的“抛锚之处” (anchor) 。
- 基于合一的特征约束 (unification-based features)

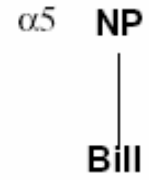
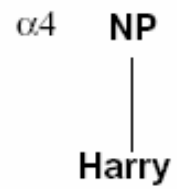
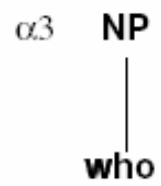
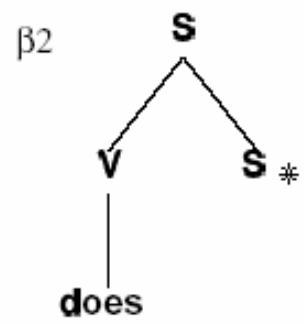
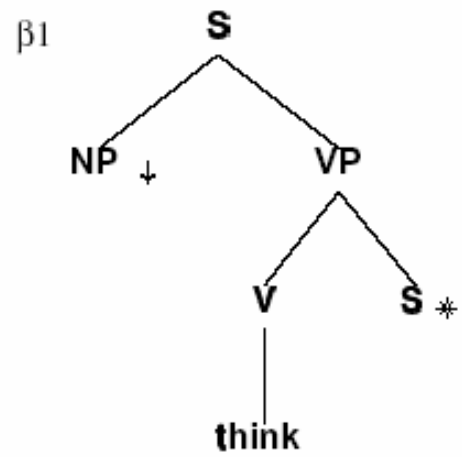
LTAG: 带锚点的树

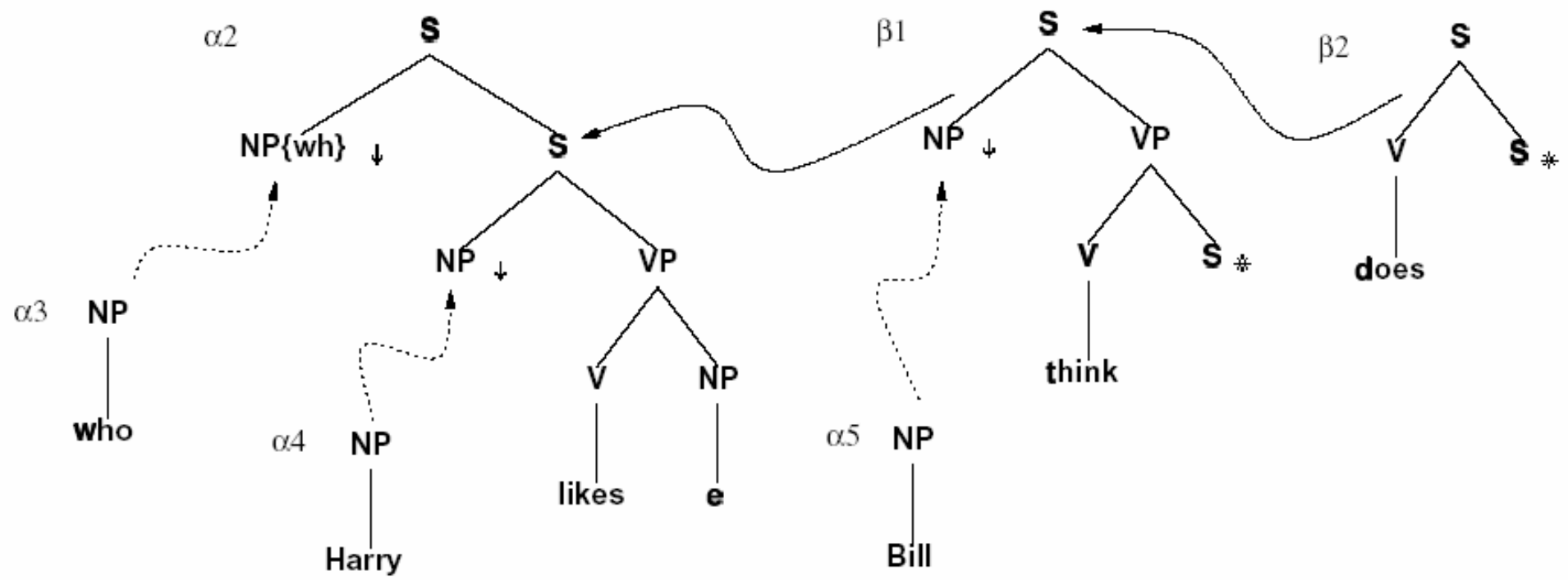


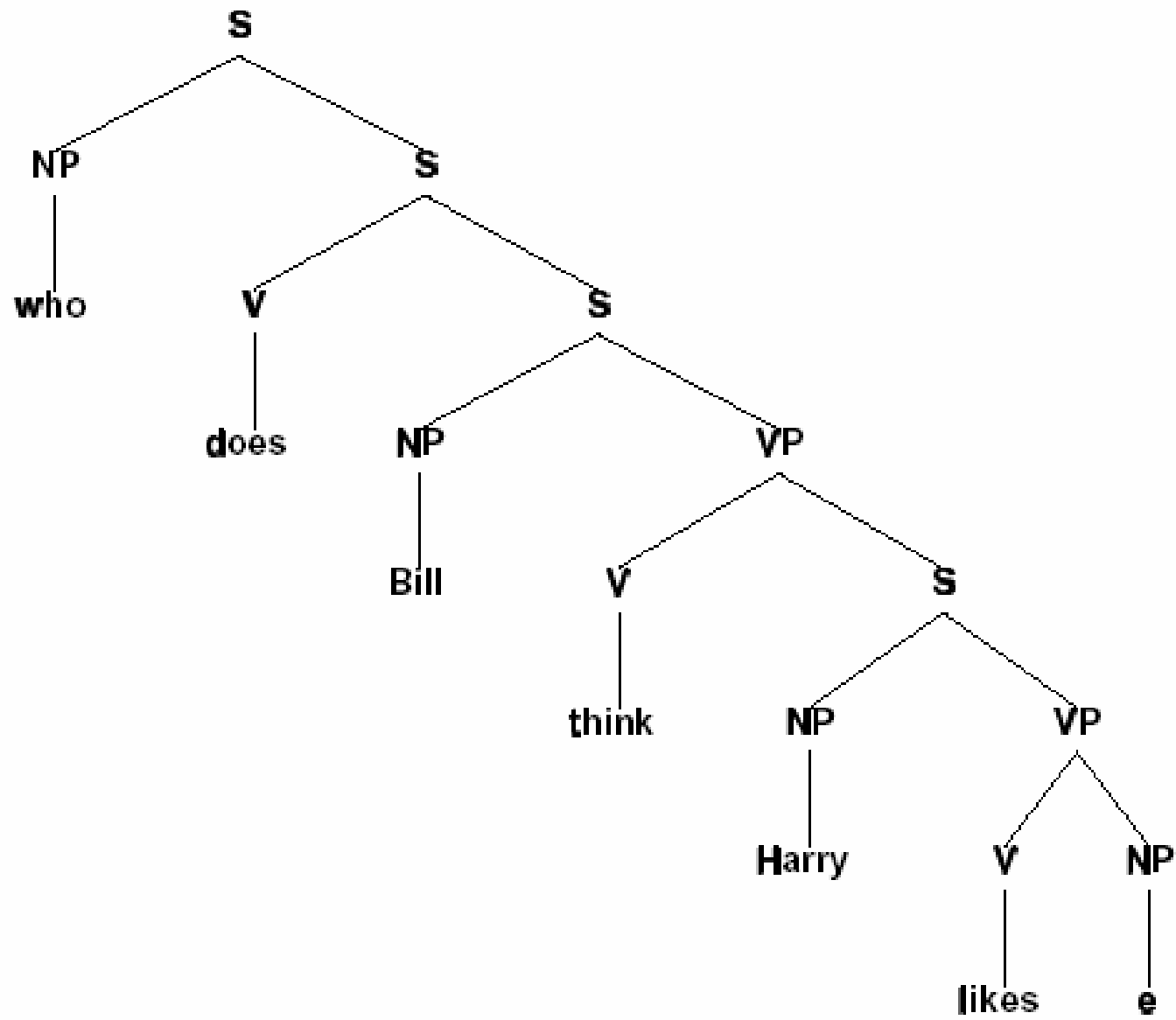
* John walked Pennsylvania.
John walked to Pennsylvania.



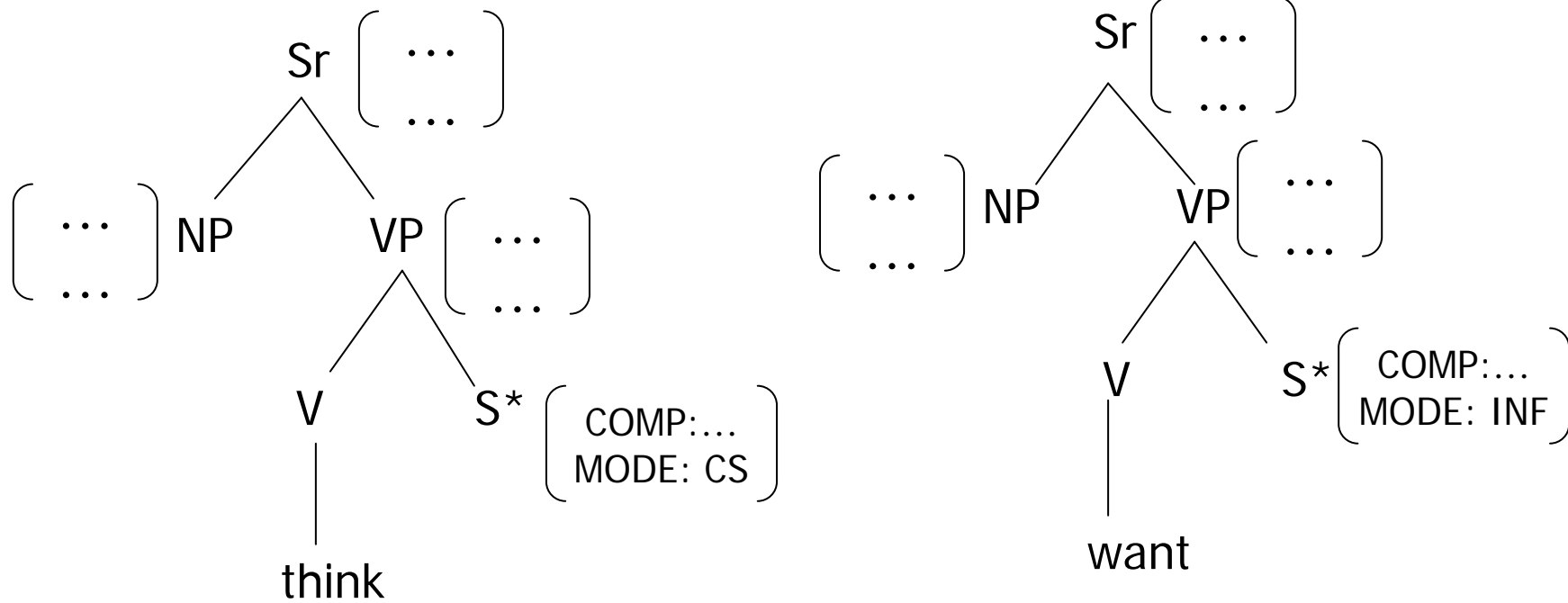
引自技术报告 (Technique Report) A Lexicalized Tree Adjoining Grammar for English, XTAG Research Group, UPenn, 2001, 2, 26







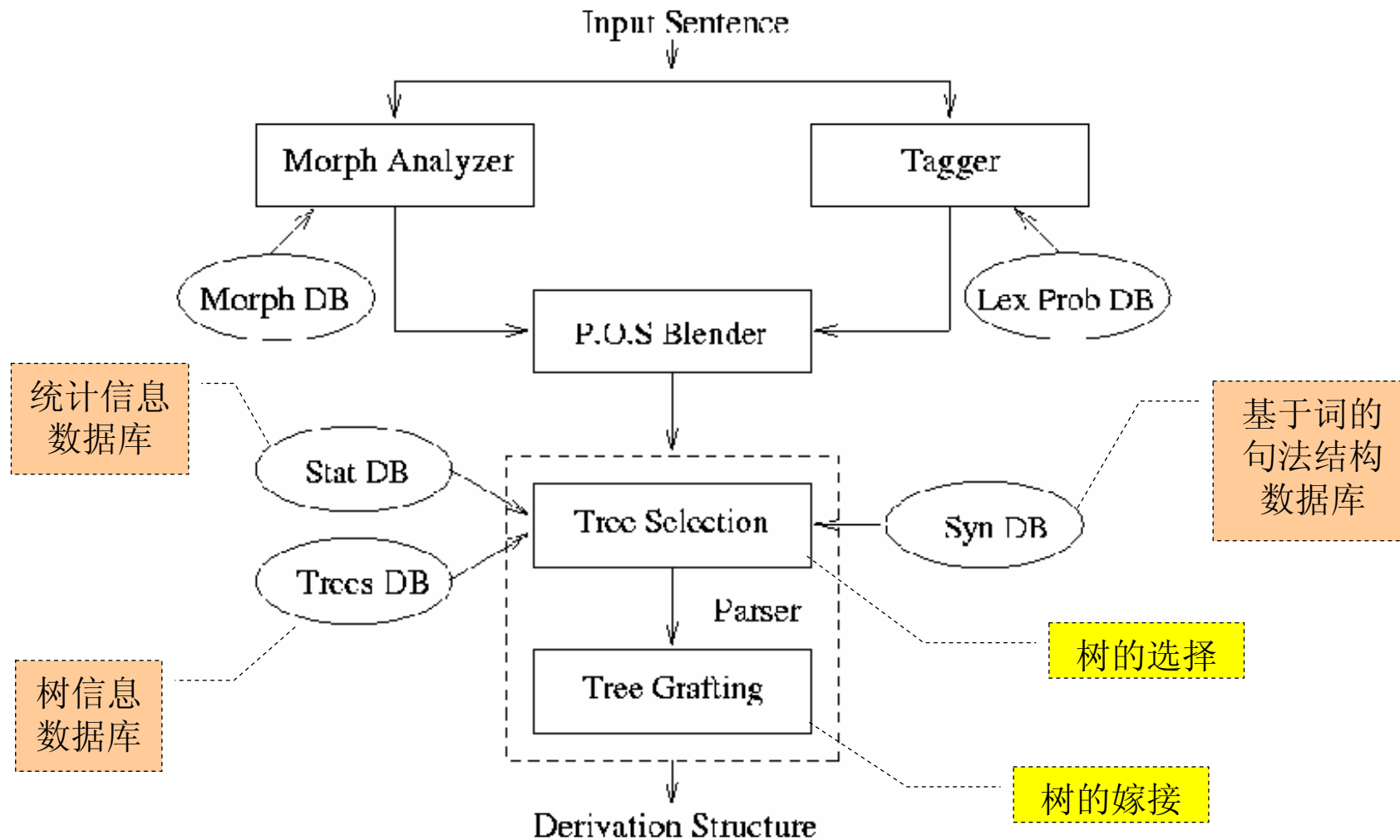
LTAG : 帶特征结构的树



John thinks that Mary loves Sally.

John wants to love Mary.

XTAG系统： LTAG理论针对英语的实践



XTAG系统的基本组成与规模

- 形态分析器与形态数据库：大约317000个词汇（含各种屈折形式），其中根词大约90000个，所以记录以屈折形式建立索引，带有根词形式和词性信息及词缀信息
- 词性标注器（POS Tagger）和句法分析器（Parser）
- 句法数据库（Syn Database）：超过31000个词，每个词包含词性信息，它所属的树结构信息和特征约束信息；
- 树数据库（Tree Database）：1004棵树，由53个树家族（Tree Family）和221个其他类型的树结构（Individual Tree）组成；

8 Link Grammar（链语法）

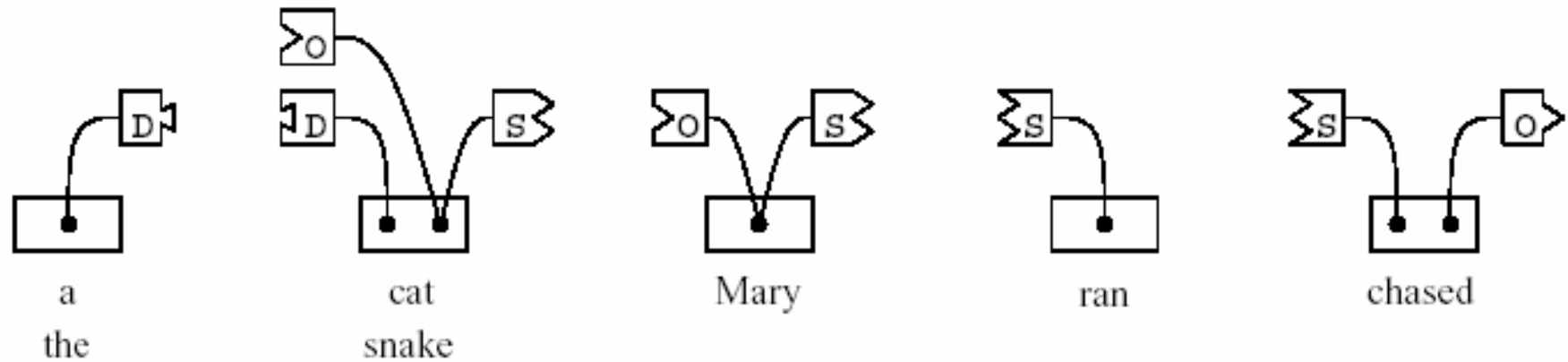
- 链语法由美国CMU（卡耐基梅隆大学）计算机学院的Daniel Sleator和美国Columbia University（哥伦比亚大学）音乐系的Davy Temperley共同提出，最早的文章是1991年的一个技术报告，题目是“Parsing English with a Link Grammar”（CMU-CS-91-196, October 1991）
- <http://www.link.cs.cmu.edu/link/>

链语法的基本特点

- 链语法（Link Grammar）不是建立在树结构的基础上，而是将语言知识完全落实到词汇基础上，通过词语的链接（Link）属性，来对句子进行分析。
- 链语法对句子的分析结果表现为句中词汇之间的链接关系，即不是“树”结构，而是“图”结构。
- 跟其他形式语法系统相比，链语法是持**强词汇主义观点**的形式语法系统。它并不强调语言成分的层次组合关系，而是从词汇的局部着眼，力图揭示：
一个句子中任意两个词之间是否有联系，以及是什么联系。

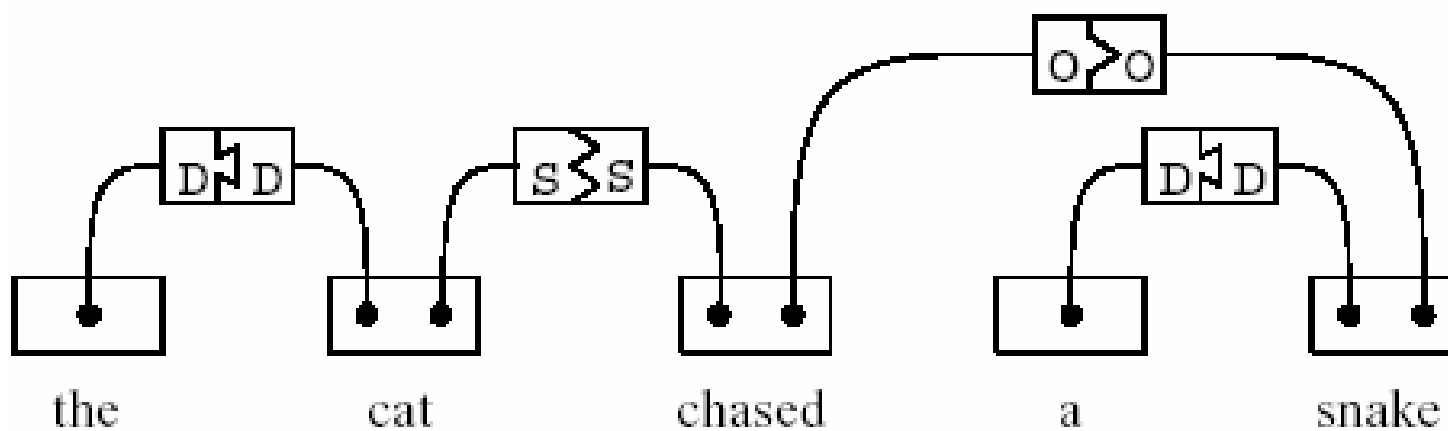
以链语法的眼光看句子的构成

- 词典中带链接的词



以链语法的眼光看句子的构成（续）

- 由词链接而成的句子



基本概念

- 一个语言的一部链语法就是该语言中的单词的集合，并且对每个单词都定义了它的**链接要求（linking requirement）**。单词的链接要求可以通过一个或若干个**链接表达式（formula of connector）**指定。

一个链语法实例

单词	链接表达式	说明
a	D+	D是链接冠词（Determiner）和名词的链
the	D+	+ 表示向右链接
cat	D- & (O- or S+)	O是链接动词和宾语（Object）的链
snake	D- & (O- or S+)	S是链接动词和主语（Subject）的链
chased	S- & O+	& 表示逻辑上的“并且”关系
ran	S-	- 表示向左链接
Mary	O- or S+	or 表示逻辑上的“或”关系
dog	{@A} & D-	@A表示该单词可以有多个A链接

链接表达式的定义

- (1) 一个链接表达式由链接子 (connector)、二元操作符 (&, or) 以及圆括号 (规定了组合符合的优先顺序) 组成。只含一个链接子的链接表达式有时简称为链接子;
- (2) 一个链接子由链名 (name) 和链接方向 (direction) 两部分组成;
- (3) 链名是个符号串, 用于标记两个单词之间的关系;
- (4) 链接方向有两个, 向左 (-) 和向右 (+);

e.g. Cat : D- & (O- or S+)

链接表达式的内部顺序

- 如果一个链接表达式由多个链接子组合而成，其中有并列关系的链接子之间是有顺序要求的，比如：“cat”的链接表达式是：

D- & (O- or S+)

就不能写成： (O- or S+) & D-

如果写成后者，就会出现先有O链，再有D链的情形。

链接子的满足

- 单词串中某个单词如果有一个向右的链接子，例如 X_+ ，而另一个单词有一个向左的链接子 X_- ，那么这两个链接子相互匹配(match)，在这两个单词之间就可以画一条 X 链。这时，我们说链接子 X_+ 或 X_- 得到了满足(Satisfaction)或说存在一个链接，满足了链接子 X_+ 或 X_- 。

链接表达式的满足

- ◆ 链接表达式 $X \& Y$ 要被满足，则链接必须同时满足链接子X和Y。
- ◆ 链接表达式 $X \text{ or } Y$ 要被满足，则链接必须至少满足链接子X和Y中的一个。

用链语法来检查句子合法性

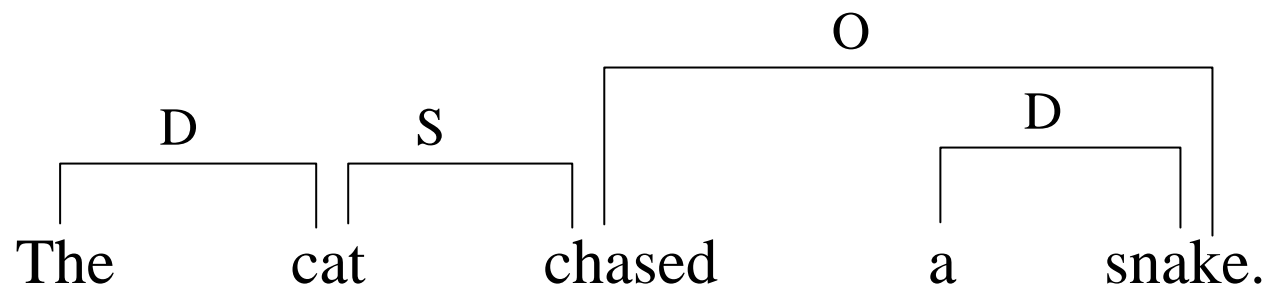
- 对于一个由单词组成的串（**S**），如果根据一部链语法（**LG**），**S**中所有单词的链接要求都得到满足且每个链接要求都只被满足一次，并且所有的链接符合下面4条元规则（**Meta Rule**）的要求：
 - ✓ 平面性（**Planarity**），链与链之间互相不交叉；
 - ✓ 连通性（**Connectivity**），所有的单词应该链在一起，形成连通图。
 - ✓ 顺序性（**Ordering**），链接表达式中靠前的链接子跟距离该单词较近的单词链接，链接表达式中靠后的链接子跟距离该单词较远的单词链接。
 - ✓ 排它性（**Exclusion**），一对单词之间不能同时有两个链接。

就说**S**是**LG**所定义的语言中的句子。使得**S**合法的全部链接称为一个链接集（**Linkage**），链接集就是链语法分析句子的结果，如同一般的**CFG**分析结果为句法树那样。

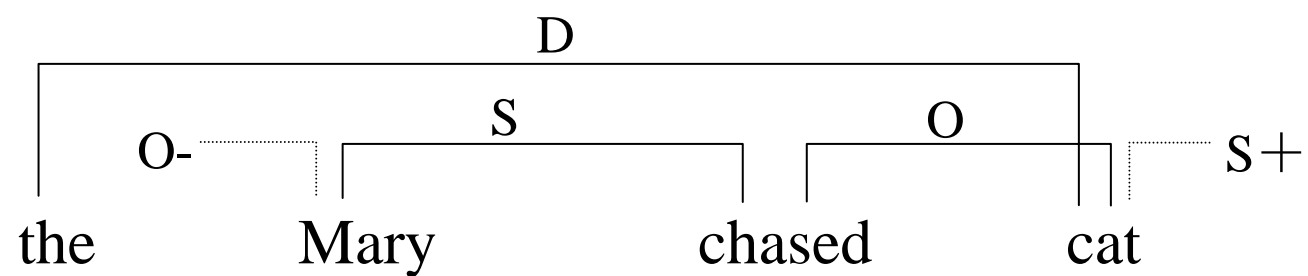
链语法的应用示例

- (1) the cat chased a snake.
- (2) * the Mary chased cat.

例1的链语法分析结果示意图



例2的链语法分析结果示意图



违反平面性

增强链语法的限制能力

对链接子进行细化

- The **black** dog died. $A+ \rightarrow Ac+$
- * The black idea died. $dog : Ac- ; idea : Aa-$

(1) the **dog** seems to have **slept** here
(2) ?The **idea** seems to have **slept** here.

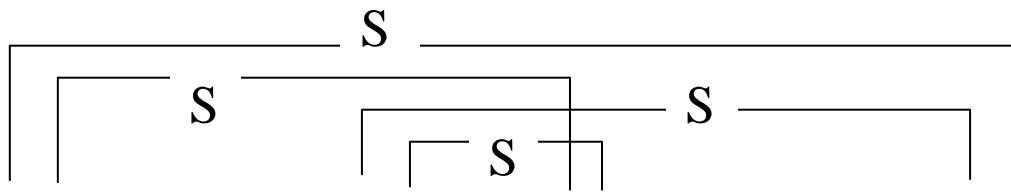
sleep : Tc- dog: Ssc+ idea : Ssa+

规则: Ssa链 与 Tc链不能同在一个 Linkage 中

dog, idea与sleep超距相关, 增加规则描述

标准链语法不易处理的语言现象

- 并列结构



Danny and Davy wrote and debugged a program.

I gave John a doll and Mary a gun. — 会被链语法拒绝

标准链语法不易处理的语言现象（续）

- 形式主语一代词it

- 1) It seemed likely that John would go.
- 2) * Joe seemed likely that John would go.
- 3) * It acted likely that John would go.

这些句子都会被链语法接受

9 Dependency Grammar（依存语法）

- 依存语法也称从属关系语法，最早是法国语言学家特尼埃尔（L. Tesniere）提出的，其主要思想反映在他去世后于1959年出版的《结构句法基础》（Element de Syntaxe Structruale）一书中。但他在发表于1934年的《怎样建立一种句法》（Comment Construire Une Syntaxe）这篇论文中就已经提出了这一语法思想。
- Haim Gaifman, 1965, Dependency systems and phrase-structure systems. Information and Control, Vol.8, No.3, pp304-337.
- Jane J. Robinson, 1970, Dependency structures and transformational rules. Language, 46:259-285.
- 冯志伟, 1983, 《特思尼耶尔从属关系语法》，载《国外语言学》1983年第1期
- Peter Hellwig, 1986, Dependency Unification Grammar, Proceedings of Coling'86, Bonn.
- 吴升（1992），周明、黄昌宁（1994）

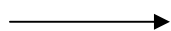
Tesnière依存语法的基本框架

- 关联（Connexion）：句子成分之间的从属关系
- 组合（Jonction）：句子的并列扩展
- 转位（Translation）：词语功能的转移

- 动词是句子的中心，动词支配其他成分，它本身不受支配；
- 直接受动词支配的有名词词组和副词词组；
- 名词词组是动词的**行动元**（actant）；
- 副词词组是动词的**状态元**（circonstant）；
- 行动元的数目就是**动词的价数**（valence）；

动词的行动元 —— 配价

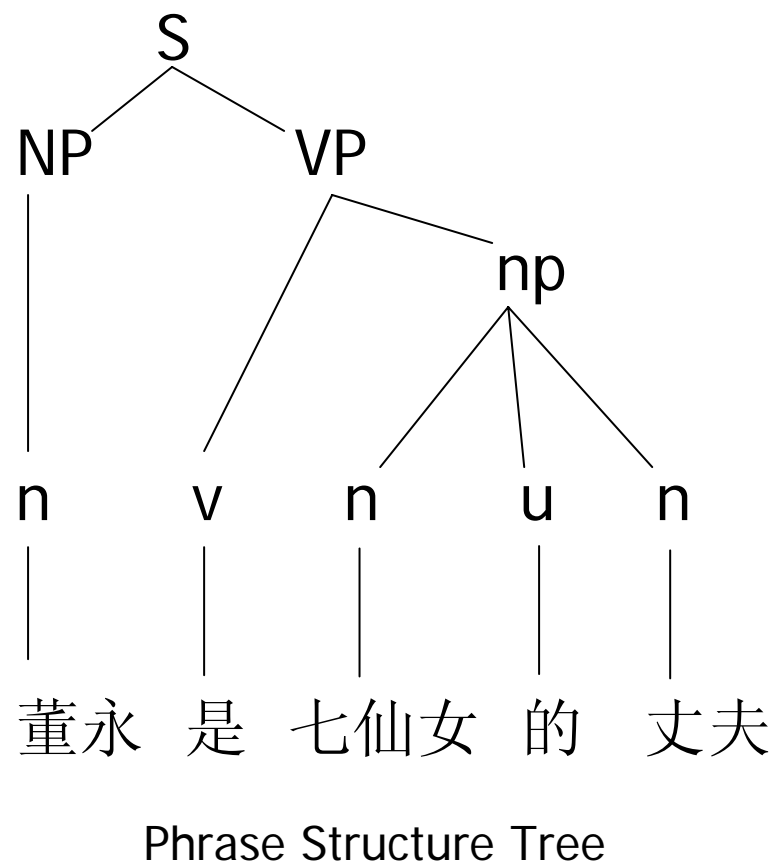
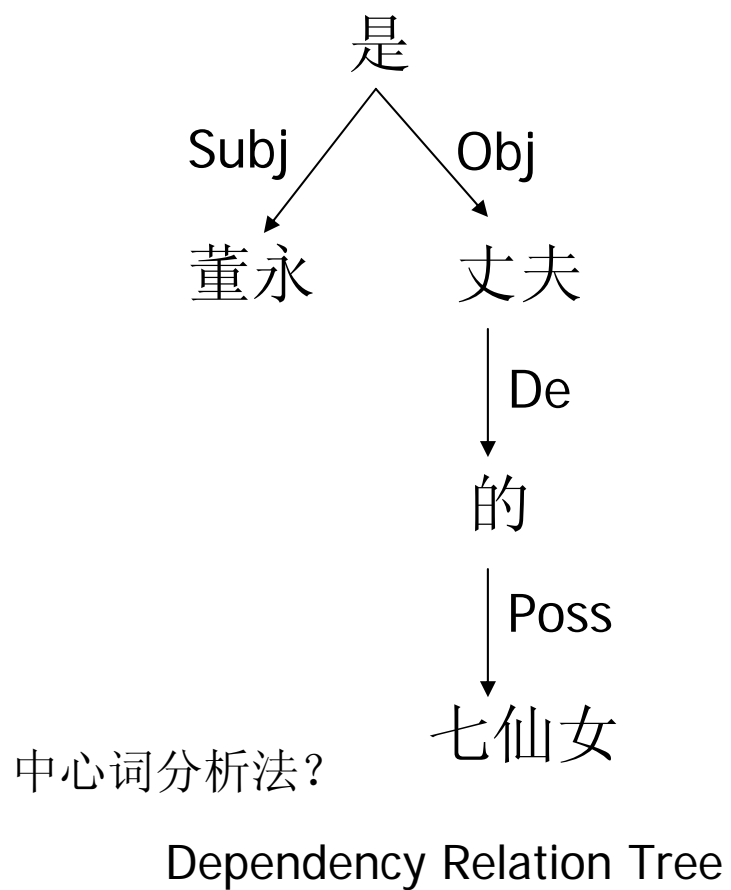
- 零价动词：不支配任何行动元, “例如”
 - 一价动词：支配一个行动元, “游泳”
 - 二价动词：支配两个行动元, “购买”
 - 三价动词：支配三个行动元, “送给”
-
- 依存语法 配价语法 (Valence Grammar)



Robinson提出的依存关系四大公理

- 1 一个句子中只有一个成分是独立的；
 - 2 除独立成分外，句子中其他成分都必须依存于某成分；
 - 3 句中任何一个成分都不能依存两个以上的其他成分；
 - 4 如果A成分从属于B成分，而C成分在句中处于A和B之间，则C成分或者从属于A，或者从属于B，或者从属于A,B之间的某个成分；
- 很显然，跟链语法一样，依存语法描述的是句子中词与词之间的直接关系，而且这种关系是有方向的（支配—从属），一般把支配词称为“主词”，把从属词称为“从词”。

依存树与短语结构树



依存树结构的线性表示

序号	单词	主词	关系类型
1	董永	2	Subj
2	是	0	Gov
3	七仙女	4	De
4	的	5	Poss
5	丈夫	2	Obj

虚构了一个节点
和关系
(Government)

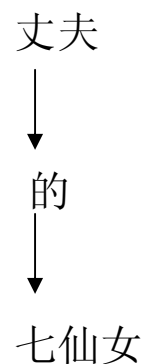
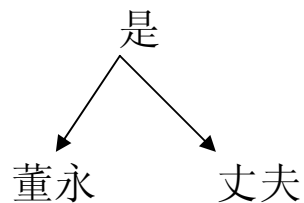
汉语依存语法的研究

- 吴升（1992），周明，黄昌宁（1994）
- 简单依存与复杂依存（自由依存与粘着依存）

三 校 联合 举办 了 这 次 竞 赛






- 传递模式（垂直同现约束）与配价模式（水平同现约束）

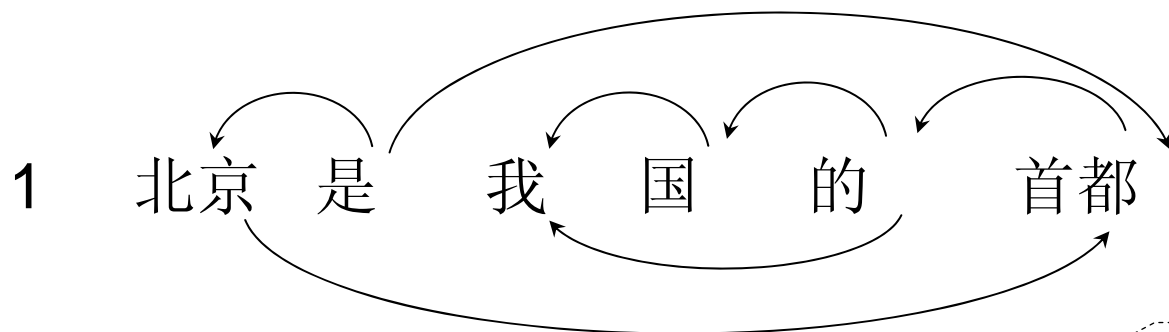


汉语的依存关系类型

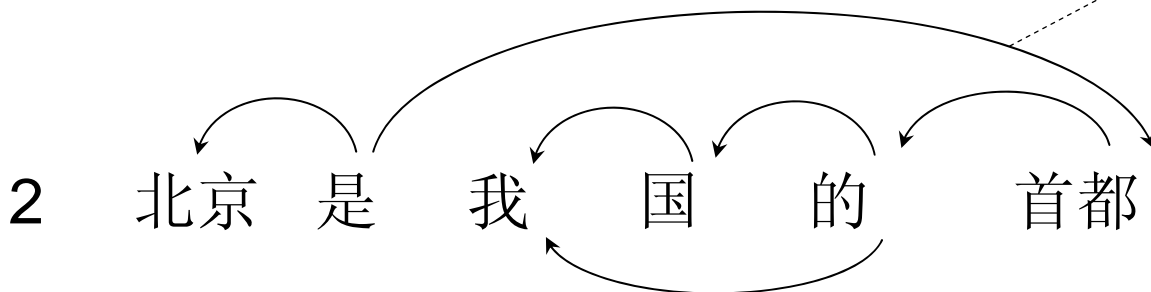
- 有多少种依存关系？（32？ 106？ 45？ 44？ ）
- 依存关系举例：宾语关系（Obj）

主词类型	从词类型	示例
V	N	我们 爱 祖国 
V	Q	我们 买 五 个 
V	De	这 本 书 不 是 我 写 的 
.....		

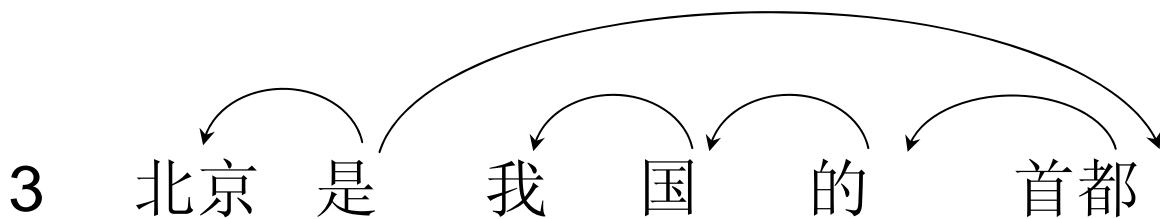
依存分析示例



错误的依存关系，
跨越了根词



错误的依存关系，跨
越了粘着依存关系；
违反公理3

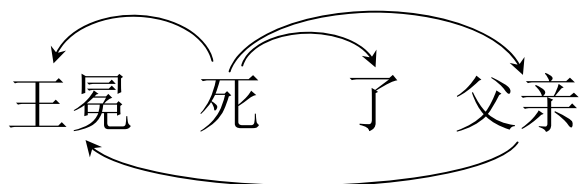


依存语法存在的问题

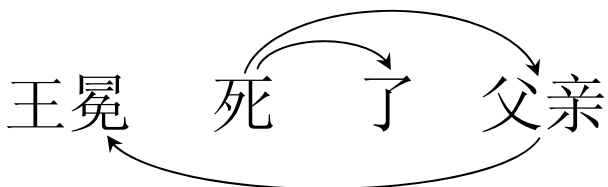
- 依存树没有反映单词之间的线性先后关系，因而在依存树的基础上进行自然语言的生成比较困难。
- 自然语言中的一些结构用依存语法表达比较困难。语义关联与句法（表层序）关联发生错位的现象。



1 合法：但语义解释不对



2 不合法：违反公理3，“王冕”有两个父结点



3 不合法：违反公理4，“王冕”从属于“父亲”，“死”位于“王冕”和“父亲”之间，但“死”不依存于其间任何一个结点

依存语法的两个发展

- 配价语法（Valency Grammar）
- 词语法（Word Grammar） —
Richard Hudson, 1984, 1990, Blackwell
- <http://www.phon.ucl.ac.uk/home/dick/wg.htm>
- 关于依存语法的更多细节内容，可以访问
<http://ufal.mff.cuni.cz/dg/dgmain.html> 查询

10 Categorical Grammar (范畴语法)

- Lesniewski, 1929, Ajdukiewicz, 1935
Bar-Hillel, 1950, Lambek, 1958, 1961,
Montague, 1970, 邹崇理, 1995, 蒋严&潘海华, 1998,
白硕, 1998
- 范畴语法的核心思想是把语言中的各种成分对应为某种“类型”/“范畴”，把语言结构的构造过程对应为“类型”/“范畴”之间的演算过程。
- <http://www.cs.man.ac.uk/ai/CG/>

基本概念

- 范畴语法里面有两种基本范畴：**S**和**N**。粗略地理解，**S**相当于"句子"，**N**相当于"名词"。
- 一个语言成分的范畴（类型）由基本范畴**S,N**加上范畴表达式构造符“\”，“/”，括号“（”，和“）”构成。
- 范畴构造符“\”表示“左缺”；“/”表示“右缺”，直观上，可以把它们设想为除号，相应地，范畴的构造就可以看成是带有方向的除法运算。括号表示结合顺序。
- 当两个语言成分之间发生结合关系时，它们相应的范畴则发生对应的“乘法”运算。运算中最关键的操作就是“约分”。

英语词类的范畴表示法

词类	范畴标注	说明
句子	S	基本范畴
名词	N	基本范畴
不及物动词	N\S	左方缺少主语
及物动词	(N\S)/N或者N/(S/N)	左方少主语，右方少宾语
形容词（做定语）	N/N	右方少中心语
形容词（做表语）	(S/N)\S	左方少“缺宾语句子”
副词（做前置状语）	(N\S)/(N\S)	右方少中心语
副词（做后置状语）	(N\S)\(N\S)	左方少中心语
介词（做后置状语）	((N\S)\(N\S))/N	右方少介词宾语
介词（做后置定语）	(N/N)/N	右方少介词宾语
冠词	N/N	右方少名词
代词（主格）	S/(N\S)	右方少不及物动词
代词（宾格）	(S/N)\S	左方少“缺宾语句子”

范畴演算

- 句子的构造过程就是语言成分所对应的范畴的演算过程。
- 一个单词串的演算的结果或者是S，或者不是S，前者即为合法的句子，后者则是不合法的句子。
- 演算的具体操作分为两种：
一种叫做“应用”(Application)，简记为A；
一种叫做“合成”(Composition)，简记为C。
- 应用就是完整的约分，即分母被约掉，只剩下分子作为结果；比如：
 $S/N \quad N \rightarrow S$ $N \quad N \backslash S \rightarrow S$
合成就是约分后范畴表达式仍然带着分母。比如：
 $S/(N \backslash S) \quad (N \backslash S)/N \rightarrow S/N$

范畴词典

he S/(N\S)

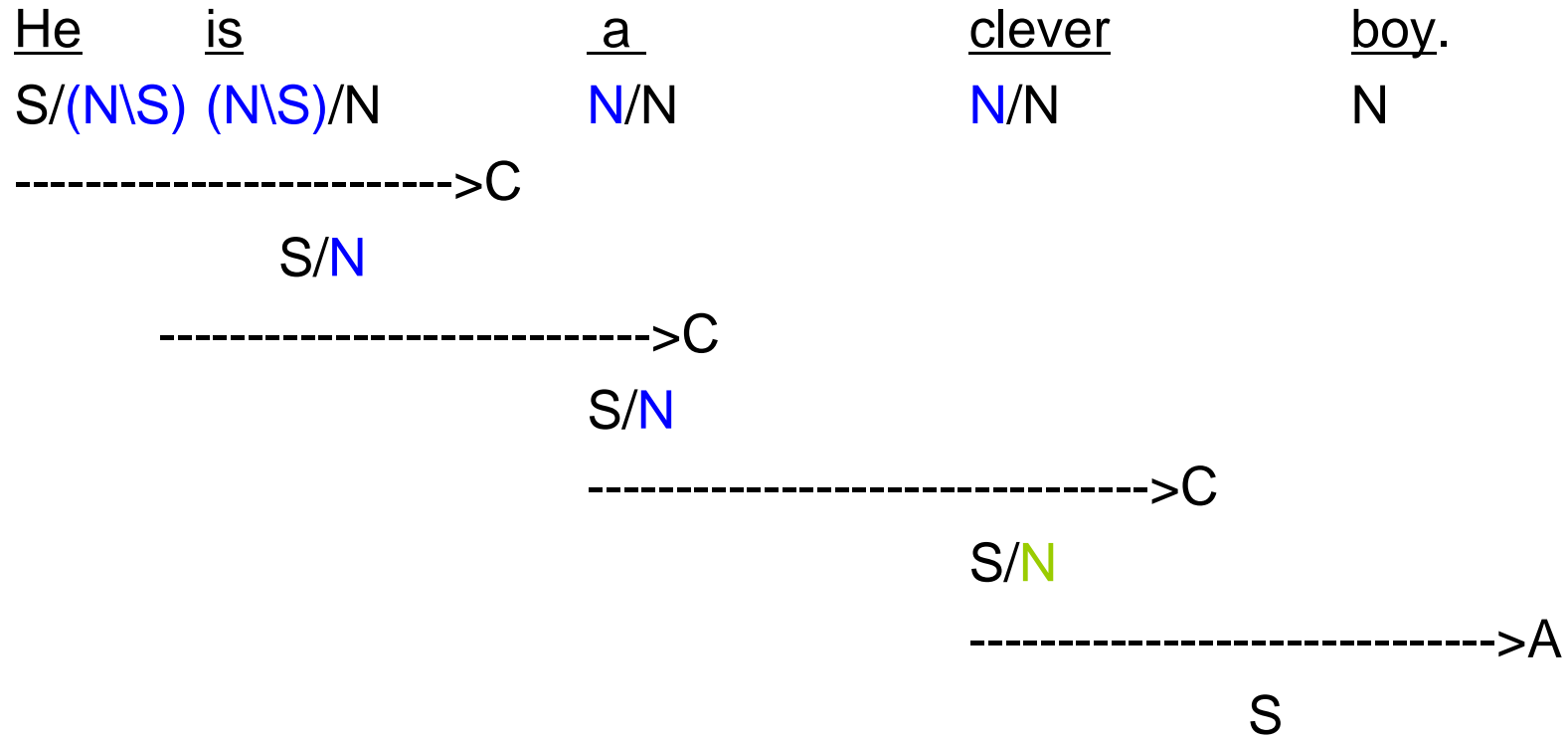
is (N\S)/N

a N/N

clever N/N

boy N

范畴演算示例



范畴演算的语言学假设

- 假设了所有结构都是由词汇负载的，这样才能从词汇的范畴标记推导出各个上级结构成分的范畴标记；
- 假设了所有结合必定是邻接成分的结合，而不可能有跨越邻接成分的超距结合，这样才能依托邻接关系实现范畴标记的约分计算；
- 假设了严格的语序关系，这样才能从确定方向上找到约分的对象；
- 假设了每个结构必须填项完备，这样才能在最后获得一个分母约干净了的**S**标记。

范畴语法存在的问题

- 1 范畴标记和词类不是一一对应的，要在具体的语流中确定具体词的范畴标记有相当的难度，甚至无异于先要理解。
- 2 不负载在词上面的结构（如汉语中的联合结构、连谓结构等）就很难纳入范畴语法的框架中去表达。
- 3 超距相关的成分（如"王冕死了父亲"中的"王冕"和"父亲"）在范畴语法的框架中无法建立约分关系。
- 4 象汉语这样语序灵活、填项经常不完备（省略）的语言，用范畴语法去描述会遇到许多麻烦问题。

参考：由短语结构树制导的范畴演算（白硕，1998）

小结

各形式语法体系之间的**共性**:

力图说明小的语言单位是如何组成大的语言单位的

各形式语法体系之间的**差异**:

- TAG与一般的短语结构语法
- 依存语法与一般的短语结构语法
- 依存语法与链语法
- 依存语法与范畴语法
- 范畴语法与链语法

这三种语法都主张直接由词来负载几乎全部语言知识

依存语法与链语法的比较

- **DG**是有向无环图，图中有一个节点是根词（**root**）；**LG**是无向有环图，图中没有根词节点；
- **Gaifman(1965)**将依存语法进行了形式化，并且论证了依存语法是上下文无关的。可以根据依存语法生成一部链语法，二者所接受的语言是一样的。

依存语法与范畴语法的比较

- 依存语法（配价语法）是对语言结构做化学类比；
- 范畴语法则是对语言结构做数学类比；
- 二者都是在描述语言成分之间的配位关系，并且假定：
一个语言成分中包含了一个或一些空位，需要另外的成分来填充这个（些）空位。
- 谁填充谁？“王冕死了父亲”
依存语法：“了”依存“死”（“了”去填充“死”）
范畴语法：“死”填充“了”（“了”是“时态结构”，
动词是“时态结构”里的一个填充物）

范畴语法与链语法的比较

范畴表达式可以转换成链接表达式。令 d 代表范畴表达式，单词 w 的所有范畴表达式为 $\{d_1, d_2, \dots, d_k\}$ ，那么与之等价的链接表达式 $E(d)$ 可以通过下面的公式转换

$$E(w) = E(d_1) \text{ or } E(d_2) \text{ or } \dots \text{ or } E(d_k)$$

上述 $E(d_k)$ 可进一步递归定义为：

$$E(f/e) = (f/e)^+ \text{ or } (f/e)^- \text{ or } (e^+ \& E(f))$$

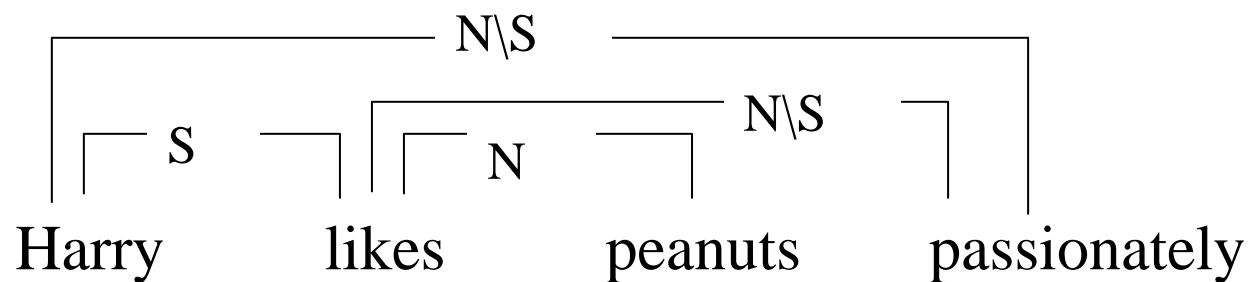
$$E(e\backslash f) = (e\backslash f)^+ \text{ or } (e\backslash f)^- \text{ or } (e^- \& E(f))$$

$$E(A) = A^- \text{ or } A^+$$

其中， f/e ， $e\backslash f$ 代表复合范畴； A 代表基本范畴

范畴向链的转换

词语	范畴表达式	链接表达式
Harry	$N, S/(N\backslash S)$	$S+ \text{ or } S- \text{ or } (N\backslash S)+ \dots$
likes	$(N\backslash S)/N$	$(N\backslash S)+ \text{ or } (S- \ \& \ (N+ \dots))$
peanuts	N	$N+ \text{ or } N-$
passionately	$(N\backslash S)\backslash(N\backslash S)$	$(N\backslash S)- \ \& \ (\dots (N\backslash S)- \dots)$



进一步阅读文献

- 吴升（1992）《汉语句法分析的语料库方法研究》，清华大学计算机系硕士学位论文，载黄昌宁、夏莹主编《语言信息处理专论》，清华大学出版社1996年版，pp171-202。
- 周明、黄昌宁（1994）《面向语料库标注的汉语依存语法体系的探讨》，载《中文信息学报》1994年第3期，pp35-52。
- 赵铁军等（2000）《机器翻译原理》，哈尔滨工业大学出版社2000年版。（第六章）
- 冯志伟（2010）《自然语言处理的形式模型》中国科学技术大学出版社。