OVERVIEW

- Introduction
- Statistical Parsing Models
  1. History-Based Models
  2. Head-Driven Models
- Results
- Future Work
- Conclusions

Parsing as a Machine Learning Problem

- Training data (the Penn WSJ Treebank (Marcus et al 93))
- Learn a model from training data
- Evaluate the model's accuracy on test data
- A standard evaluation:

  Train on 40,000 sentences from Wall Street Journal

  Test on 2,300 sentences

A Key Problem: Examples of Ambiguity

- Prepositional phrase attachment

  I (saw the man) with the telescope

  I saw (the man with the telescope)

- Part-of-speech ambiguity

  V $\Rightarrow$ saw
  N $\Rightarrow$ saw (used to cut wood...)

- Coordination

  a program to promote safety in ((trucks) and minivans)

  a program to promote ((safety in trucks) and minivans)
  ((a program to promote safety in trucks) and minivans)

STILL MORE PARSES...

a program to promote safety in trucks and minivans

- Need a rule $\texttt{NP} \rightarrow \texttt{NP NP}$

  Suddenly Reagan the actor became Reagan the president

- *a program to promote* is an NP

- *safety in trucks and minivans* has two readings as an NP

TWO QUESTIONS

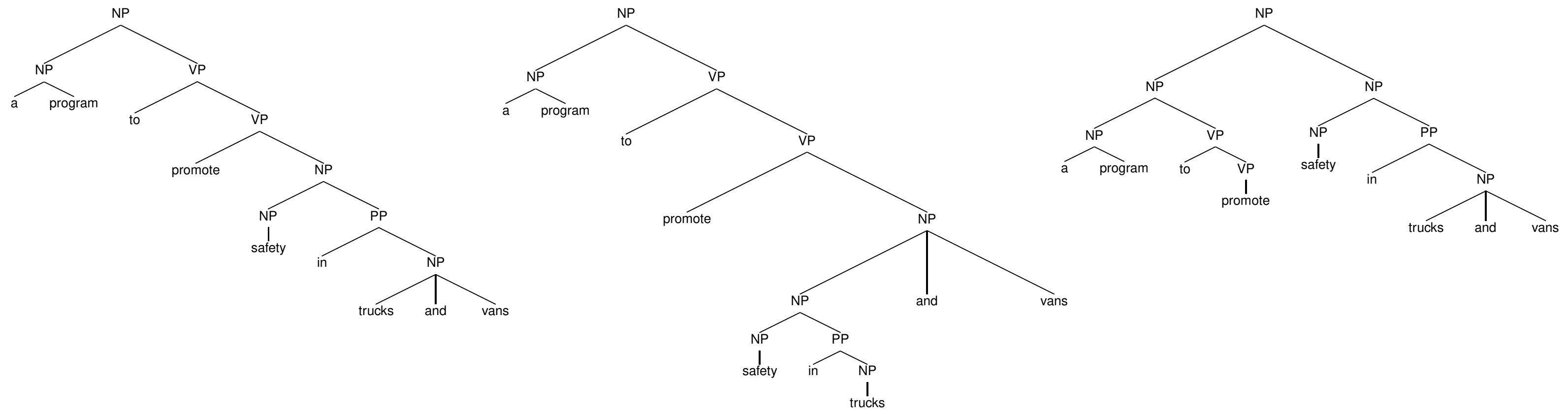
1. What objects to count?

   $Count(\text{NP} \rightarrow \text{NP NP})$, $Count(\text{program is a noun})$,
   $Count(\text{promote=transitive})$, $Count(\text{trucks, vans coordinated})$

2. How to combine the counts to give a $Score$ for each parse?

---

a program to promote safety ... $\Rightarrow$



Probabilistic Parsing

- S = a sentence.
- T = a parse tree for the sentence.
- A statistical model defines $P(T \mid S)$.
- The best parse is then

$$\begin{aligned} T_{best} &= \arg\max_T P(T \mid S) \\ &= \arg\max_T \frac{P(T,S)}{P(S)} \\ &= \arg\max_T P(T,S) \end{aligned}$$

TWO PROBLEMS

1. How to define the function which maps $(T, S) \rightarrow [0, 1]$.
   - What to count?
   - How to combine the counts?

2. Given a sentence S, how to find the tree T_{best} which maximizes $P(T, S)$?

Motivation for Lexicalization

- PCFGs give 72% accuracy: Poor use of lexical information

- Prepositional Phrase Attachment
(Hindle and Rooth 91, Ratnaparkhi et al 94,
Brill and Resnik 94, Collins and Brooks 95)

  Binary Classification:
  "saw, man, with, telescope" $\Rightarrow$ Noun or Verb-attach

Method	Accuracy
Always noun attachment	59%
P(Noun-attach $\mid$ saw,man,with,telescope)	84.1%

A General Approach: History-Based Models (Black et. al 92)

1) Representation Choose non-terminal labels, parts-of-speech etc.

2) Decomposition Define a one-to-one mapping between parse trees (T, S) and decision sequences $\langle d_1, d_2, ..., d_n \rangle$

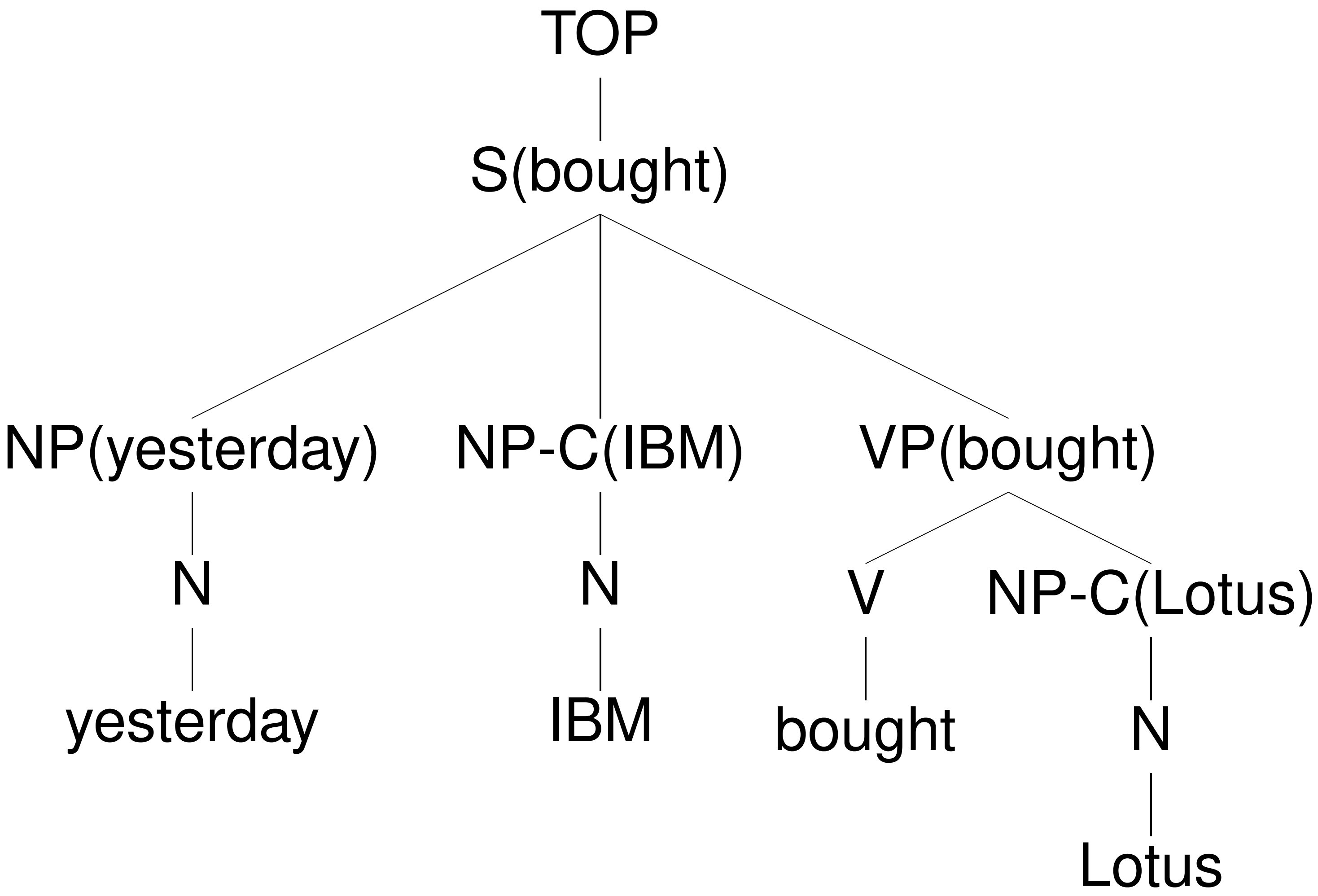
$$P(T, S) = \prod_{i=1...n} P(d_i | d_1 ... d_{i-1})$$

3) Independence Assumptions Define a function ϕ

$$P(T, S) = \prod_{i=1...n} P(d_i | \phi(d_1 ... d_{i-1}))$$

A HEAD-DRIVEN APPROACH: REPRESENTATION

Lexicalized trees

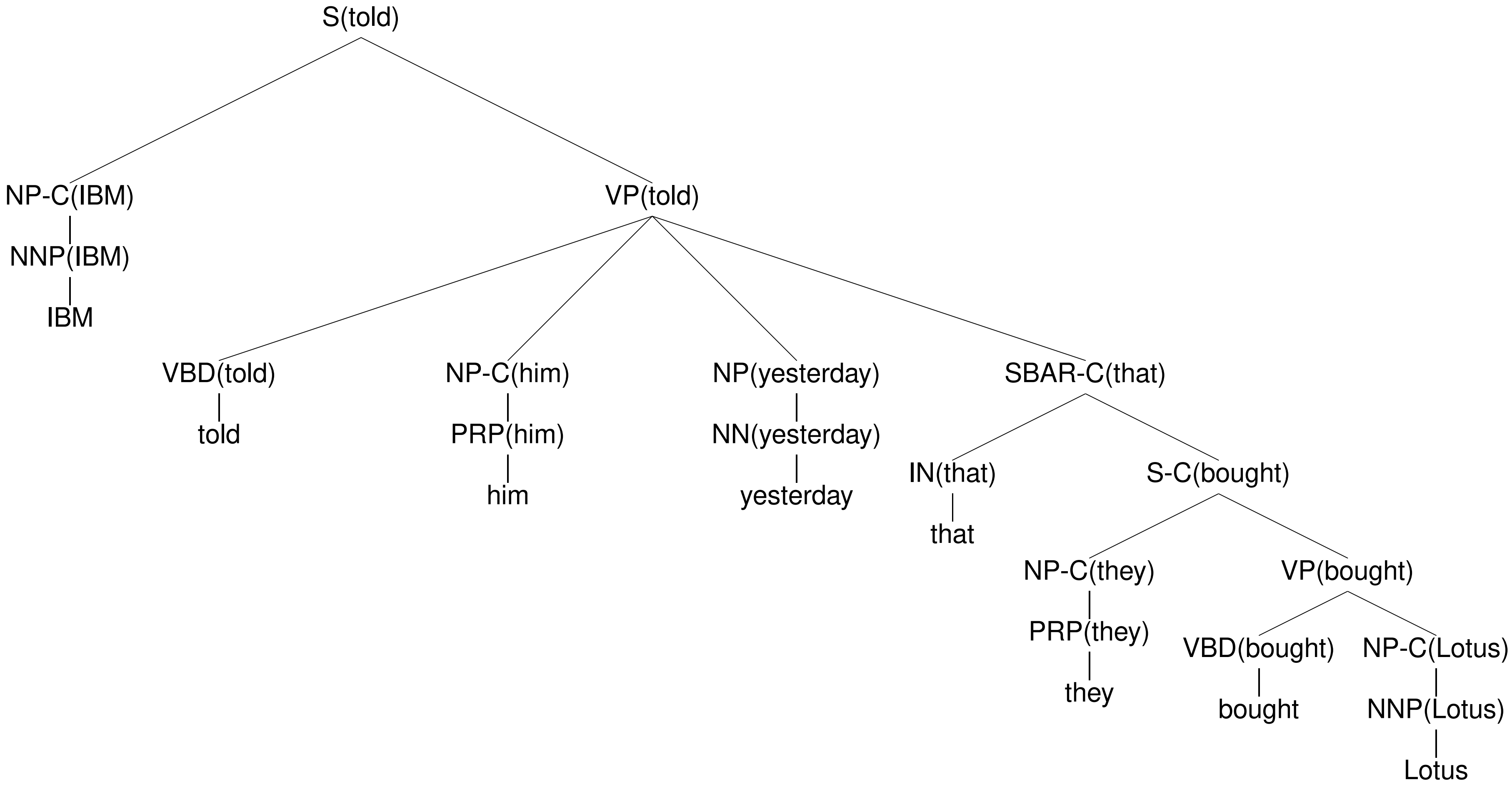


A Head-Driven Approach

Decomposition: A head-centered, top-down derivation

Independence Assumptions:

- Each parameter is conditioned on a lexical item
- Each word has an associated sub-derivation, and an associated set of probabilities:
  - Head-projection
  - Subcategorization
  - Placement of complements/adjuncts
  - Lexical dependencies


S(told)
NP-C(IBM)
NNP(IBM)
IBM
VP(told)
VBD(told)
told
NP-C(him)
PRP(him)
him
NP(yesterday)
NN(yesterday)
yesterday
SBAR-C(that)
IN(that)
that
S-C(bought)
NP-C(they)
PRP(they)
they
VP(bought)
VBD(bought)
bought
NP-C(Lotus)
NNP(Lotus)
Lotus

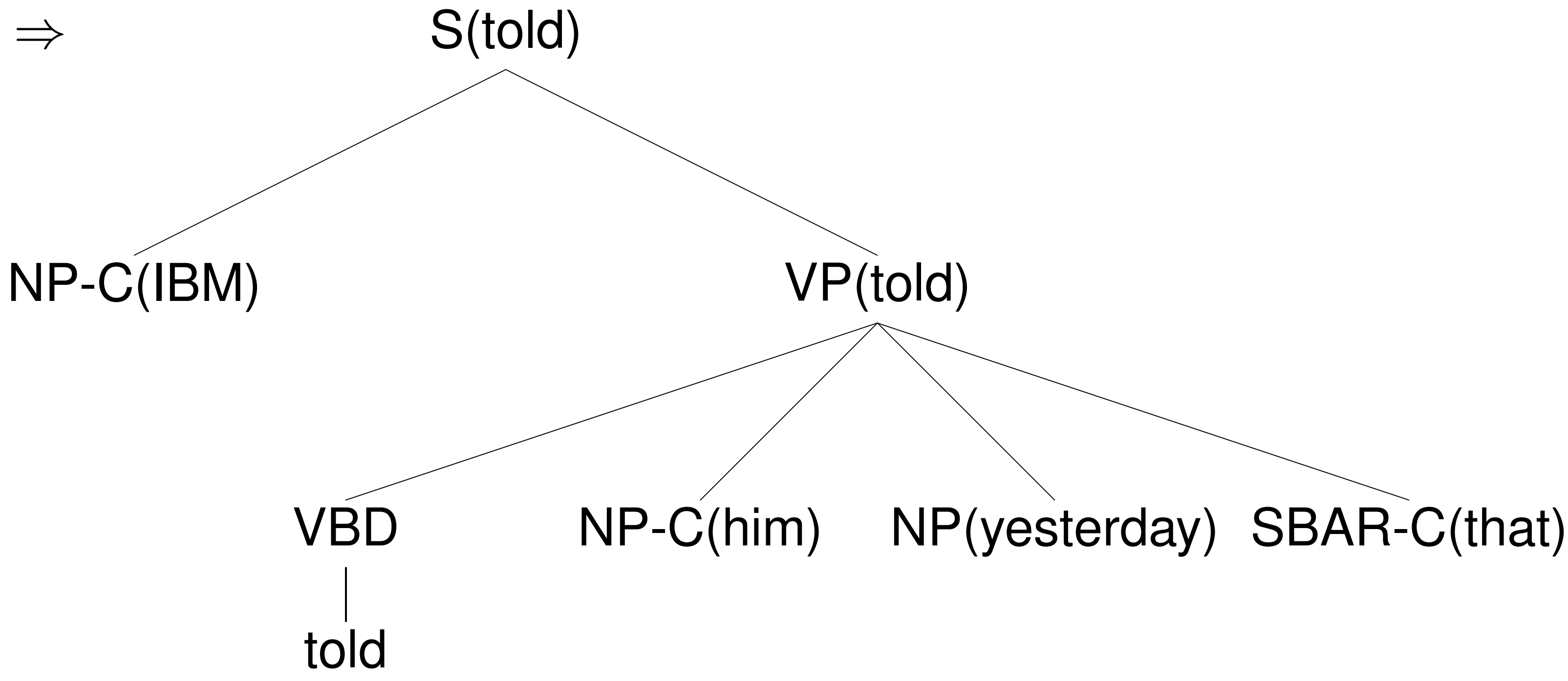
THE FIRST STEP OF THE DERIVATION

$$\text{START} \quad \Rightarrow \quad \text{S(told)}$$

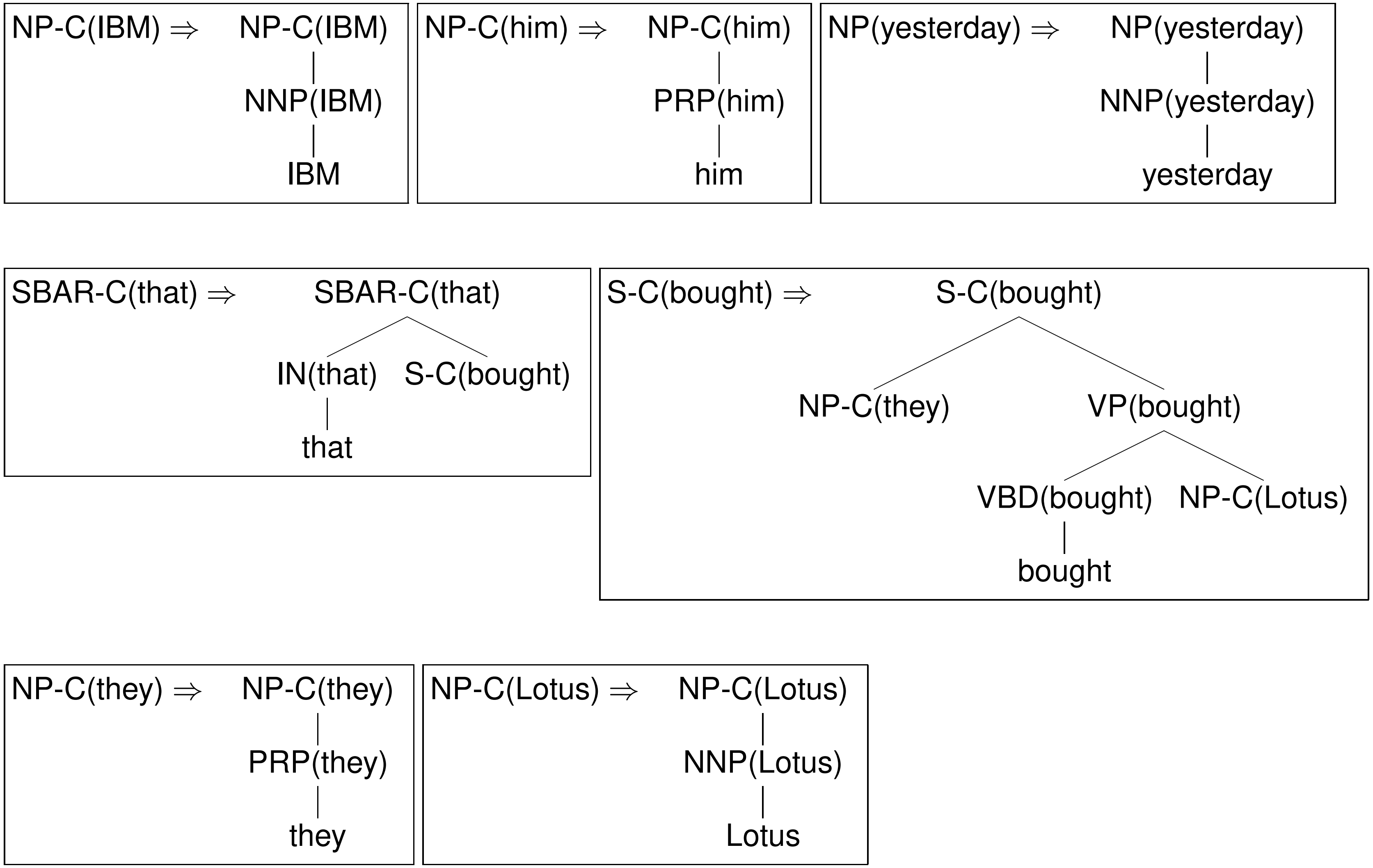
$$P(\text{S(told)}|\text{START})$$

THE SUB-DERIVATION ASSOCIATED WITH *told*

S(told) ⇒



SUB-DERIVATIONS FOR THE OTHER WORDS

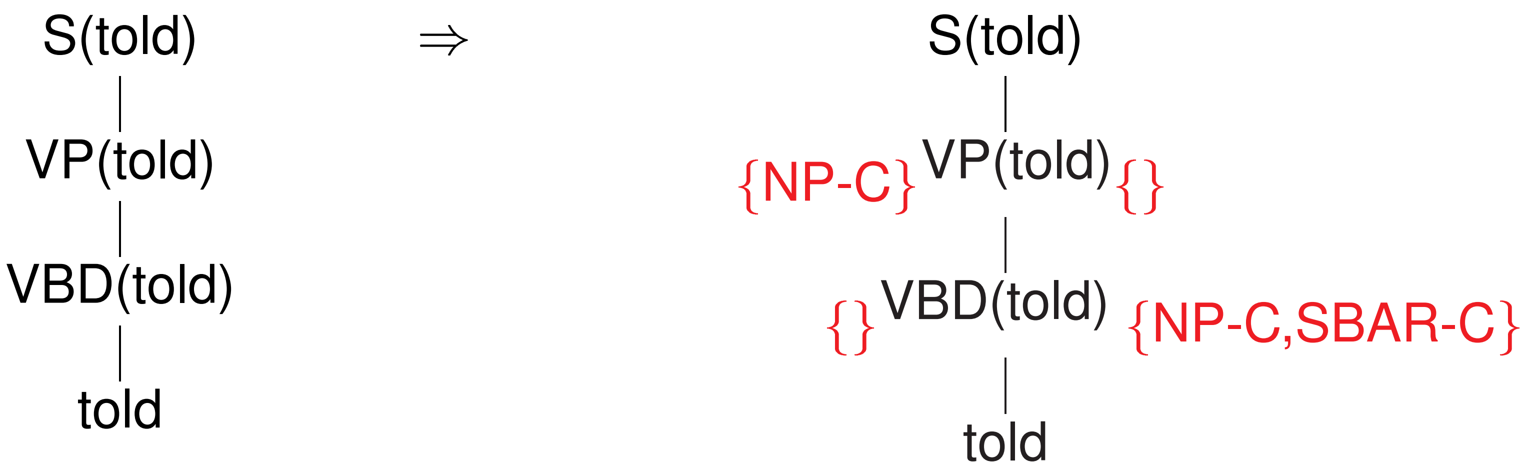


Head-Projection Parameters

$$\text{S(told)} \quad \Rightarrow \quad \begin{array}{c}\text{S(told)}\\ |\\ \text{VP(told)}\end{array} \quad \Rightarrow \quad \begin{array}{c}\text{S(told)}\\ |\\ \text{VP(told)}\\ |\\ \text{VBD(told)}\\ |\\ \text{told}\end{array}$$

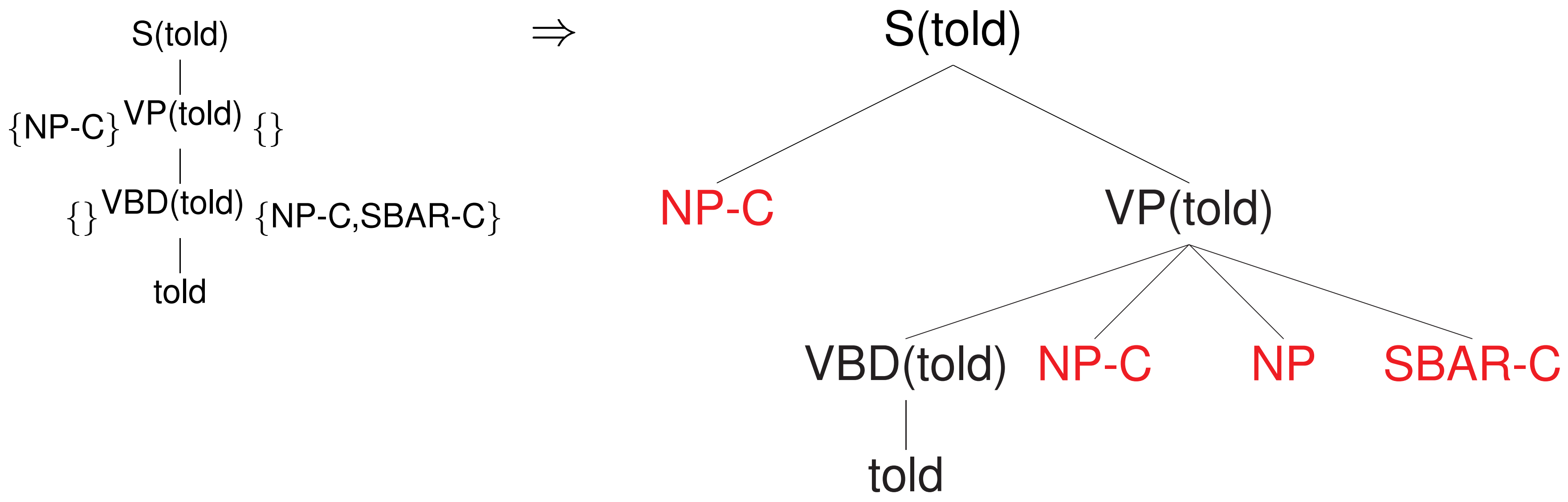
$$P(\text{VP} \mid \text{S,told}) \times P(\text{VBD} \mid \text{VP,told})$$

Subcategorization Parameters

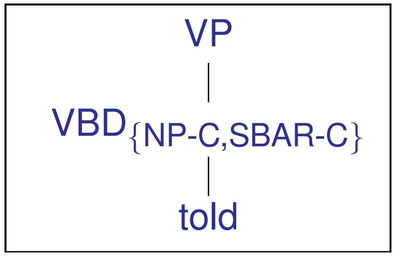


$$P(\{\text{NP-C}\}|\text{S,VP,told,LEFT}) \times P(\{\}|\text{S,VP,told,RIGHT}) \times$$
$$P(\{\}|\text{VP,VBD,told,LEFT}) \times P(\{\text{NP-C,SBAR-C}\}|\text{VP,VBD,told,RIGHT})$$

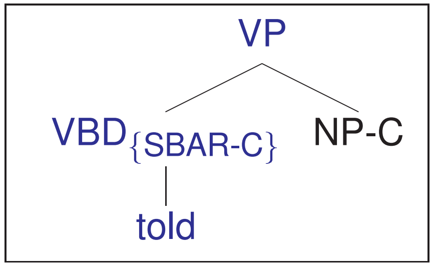
Placement of Complements and Adjuncts



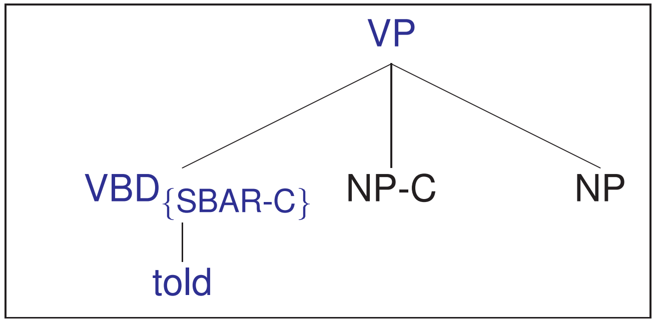
Placement of Complements and Adjuncts

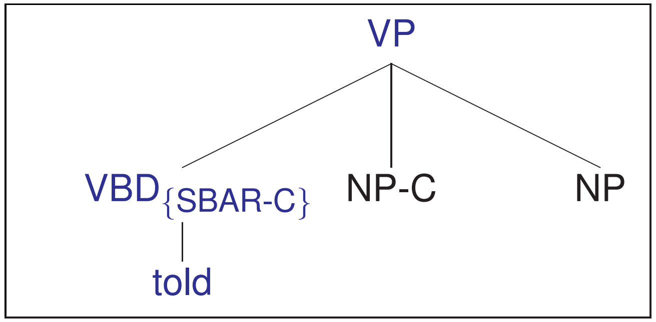


$$\Downarrow P(\text{NP-C} \mid \text{VP, VBD, \{NP-C,SBAR-C\}, told, RIGHT})$$

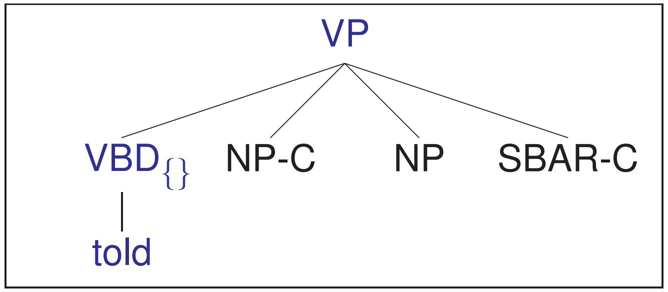


$$\Downarrow P(\text{NP} \mid \text{VP, VBD, \{SBAR-C\}, told, RIGHT})$$

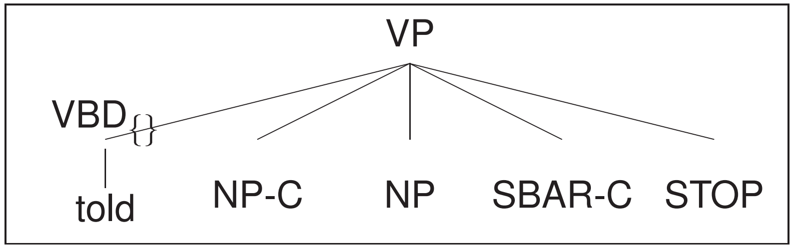




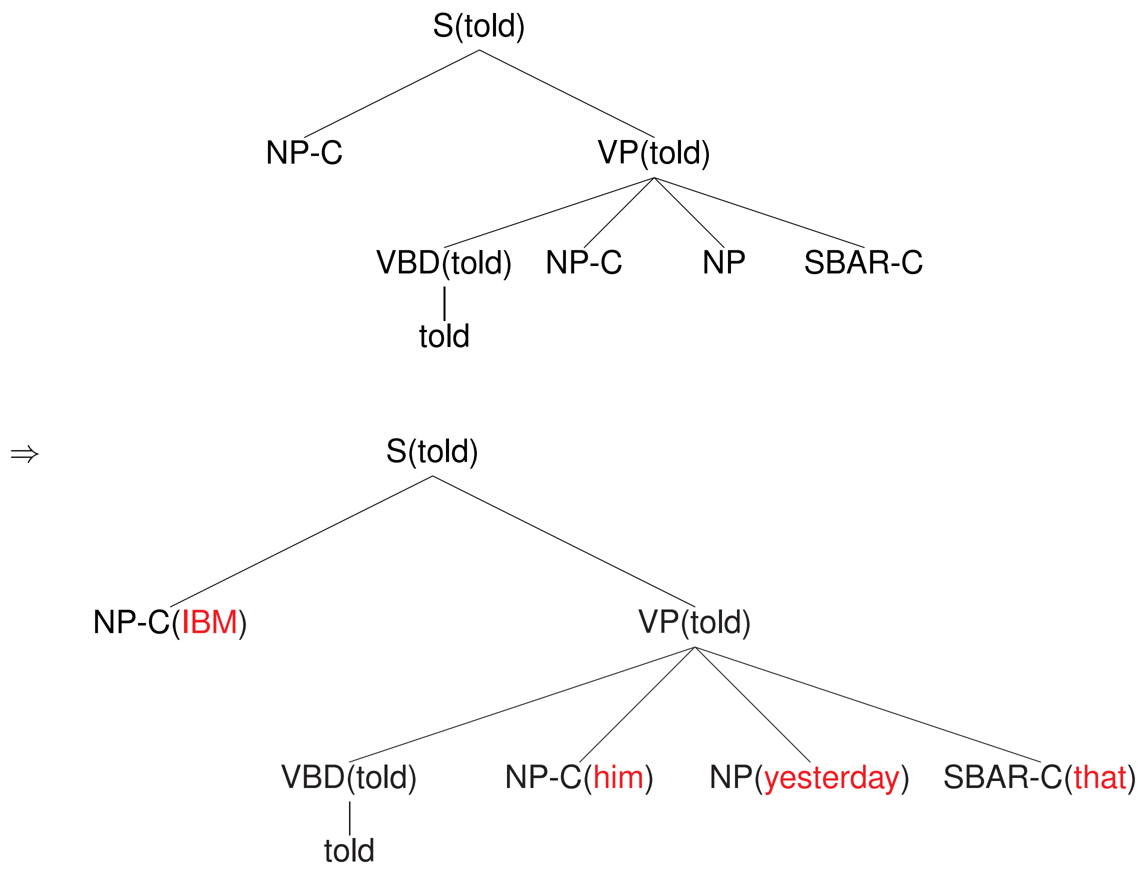
$$\Downarrow P(\text{SBAR-C}|\text{VP, VBD, } \{\text{SBAR-C}\}\text{, told, RIGHT})$$



$$\Downarrow P(\text{STOP}|\text{VP, VBD, } \{\}\text{, told, RIGHT})$$



Dependency Parameters



$$P(\text{IBM}|\text{told,S,VP,NP-C,left}) \times P(\text{him}|\text{told,VP,VBD,NP-C,right}) \times$$

$$P(\text{yesterday}|\text{told,VP,VBD,NP,right}) \times P(\text{that}|\text{told,VP,VBD,SBAR-C,right})$$

Estimation

- Maximum-Likelihood estimates:

$$P(\{\text{NP-C,SBAR-C}\}|\text{VP,VBD,told,RIGHT}) =$$

$$\frac{\text{Count}(\{\text{NP-C,SBAR-C}\}, \text{VP,VBD,told,RIGHT})}{\text{Count}(\text{VP,VBD,told,RIGHT})}$$

- Smoothing:

$$P(\{\text{NP-C,SBAR-C}\}|\text{VP,VBD,told,RIGHT}) =$$

$$\lambda \times \frac{\text{Count}(\{\text{NP-C,SBAR-C}\}, \text{VP,VBD,told,RIGHT})}{\text{Count}(\text{VP,VBD,told,RIGHT})} +$$

$$(1 - \lambda) \times \frac{\text{Count}(\{\text{NP-C,SBAR-C}\}, \text{VP,VBD,RIGHT})}{\text{Count}(\text{VP,VBD,RIGHT})}$$

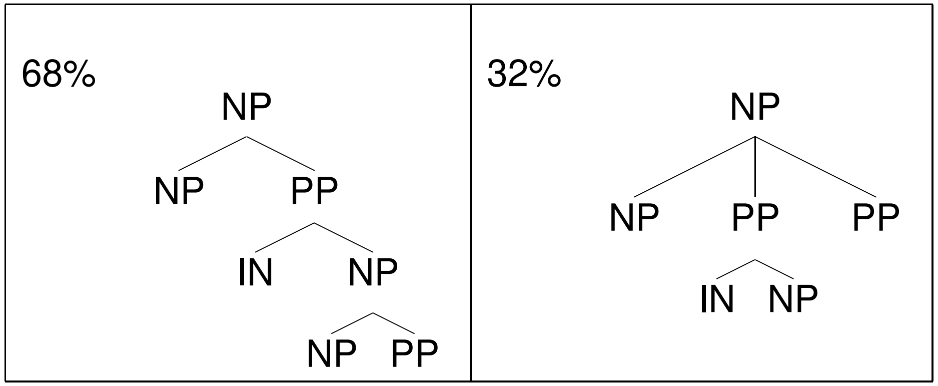
$$P(\text{him} \mid \text{told,VP,VBD,NP-C/}\textsc{prp}) =$$

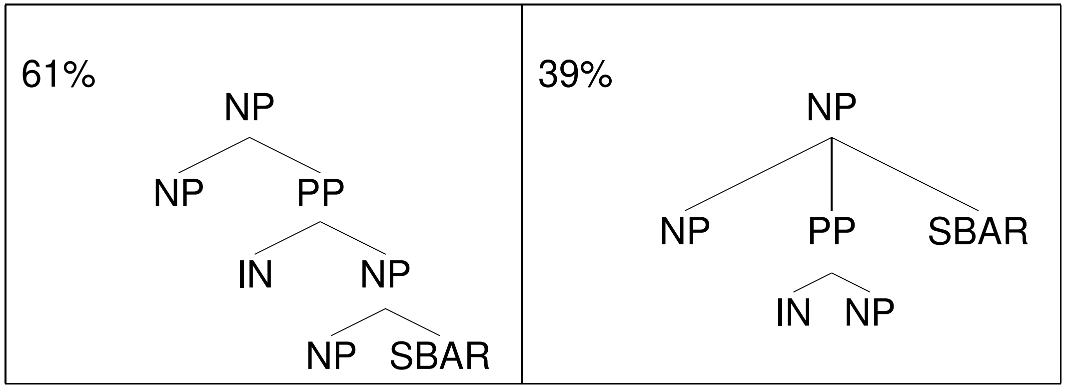
$$\lambda_1 \times \frac{\text{Count(him, told,VP,VBD,NP-C/}\textsc{prp}\text{,RIGHT)}}{\text{Count(told,VP,VBD,NP-C/}\textsc{prp}\text{,RIGHT)}} +$$

$$\lambda_2 \times \frac{\text{Count(him, VP,VBD,NP-C/}\textsc{prp}\text{,RIGHT)}}{\text{Count(VP,VBD,NP-C/}\textsc{prp}\text{,RIGHT)}} +$$

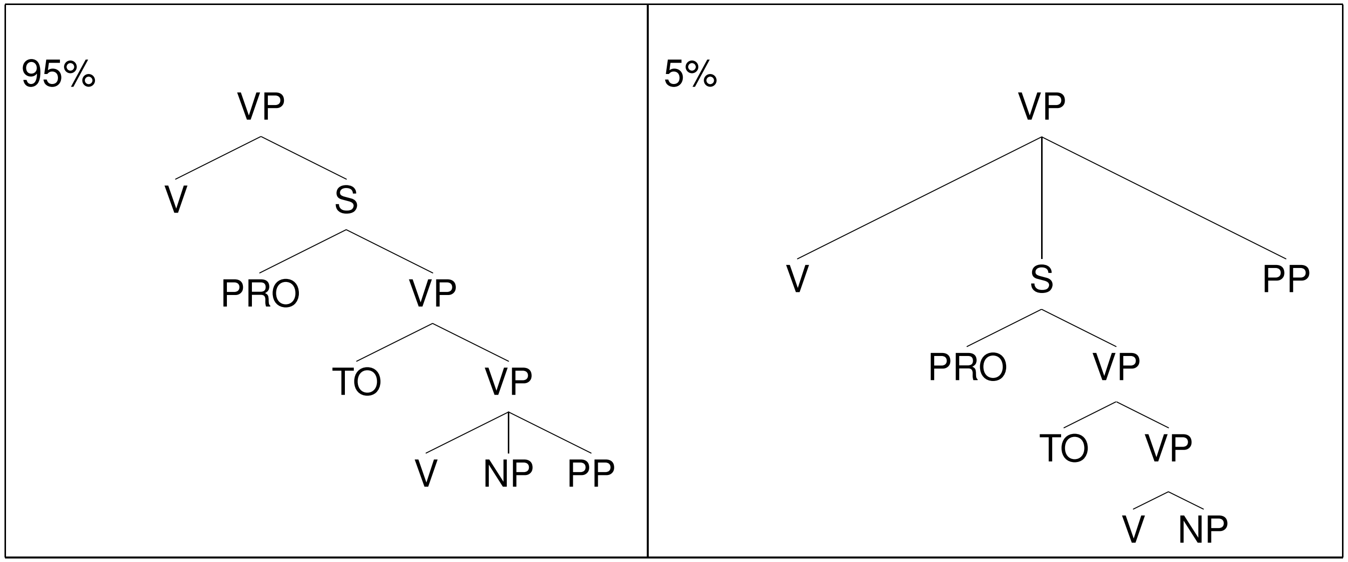
$$\lambda_3 \times \frac{\text{Count(him, PRP)}}{\text{Count(PRP)}}$$

Close-Attachment Preferences: Adjacency

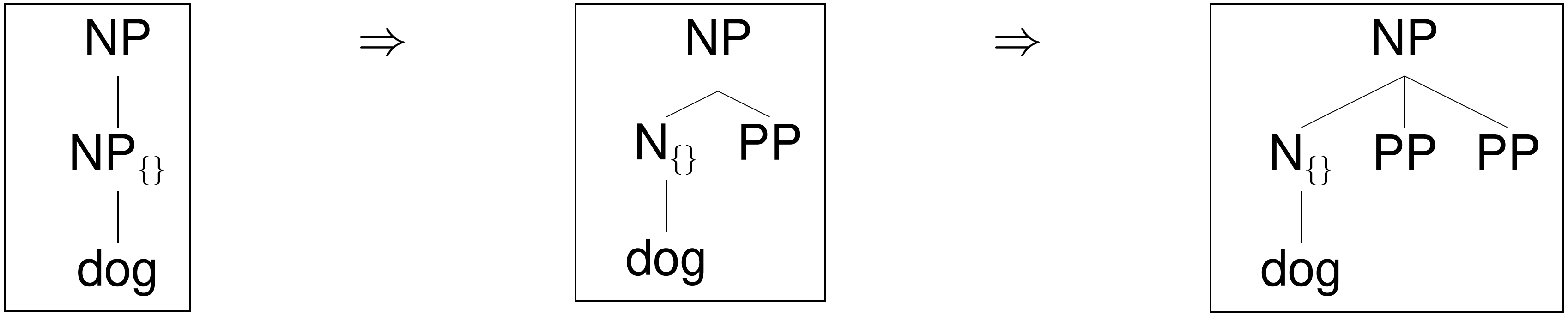




Close-Attachment Preferences: Verb-Crossing



Placement of Complements and Adjuncts: Adjacency



$P(\text{PP}|\text{NP, N, }\{\}\text{, dog, adjacency=TRUE})$

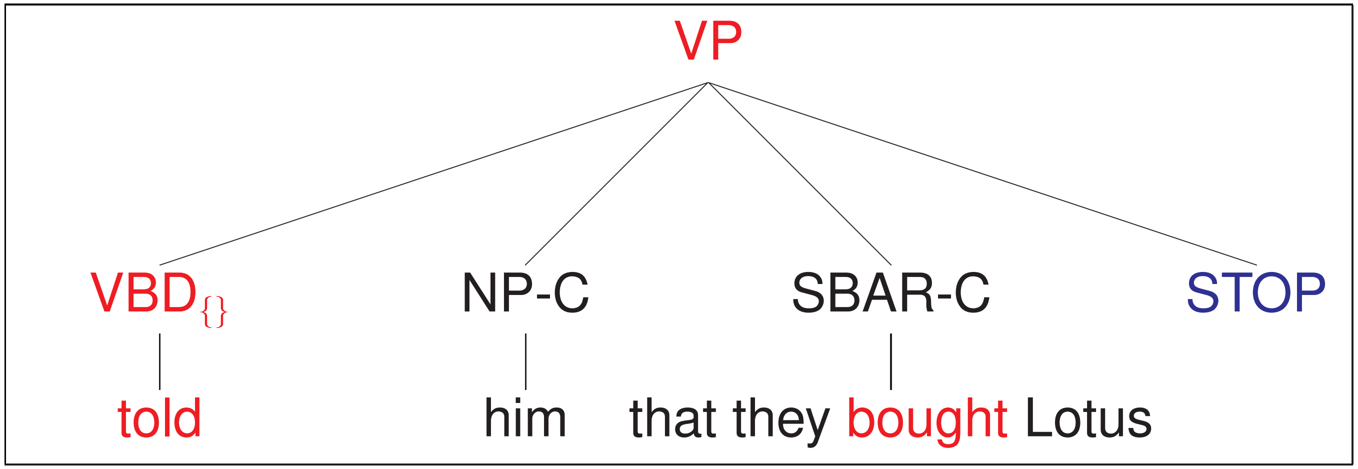
$P(\text{PP}|\text{NP, N, }\{\}\text{, dog,adjacency=FALSE})$

Close-attachment means

$$P(\text{PP}|\text{NP, N, }\{\}\text{, dog, adjacency=TRUE}) \quad > \\ P(\text{PP}|\text{NP, N, }\{\}\text{, dog, adjacency=FALSE})$$

Placement of Complements and Adjuncts: Verb-Crossing

IBM told him that they bought Lotus yesterday

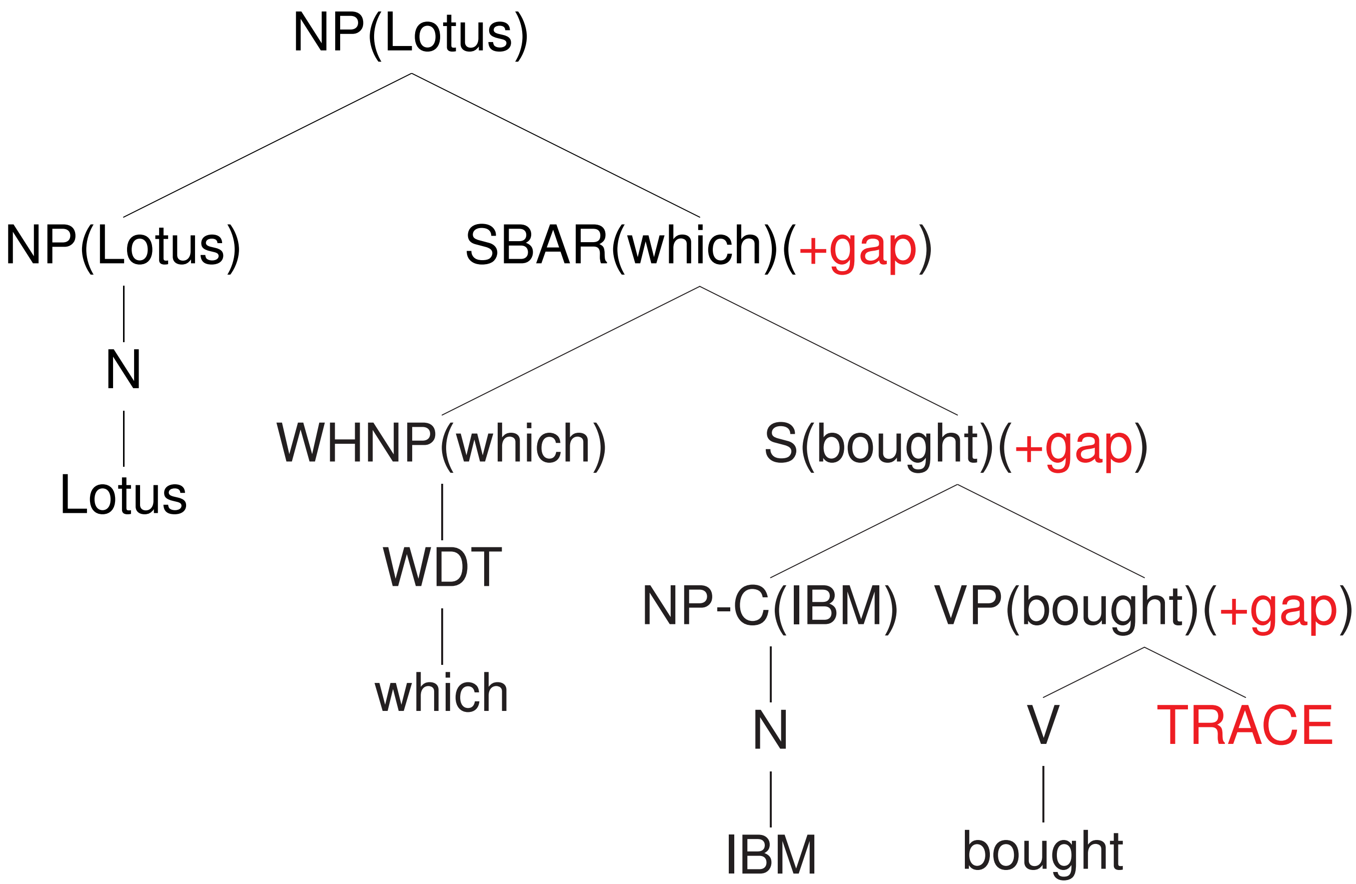


$$P(\text{STOP}|\text{VP, VBD, } \{\}, \text{ told, verb-crossing=TRUE})$$

Close-attachment means

$$P(\text{STOP}|\text{VP, VBD, } \{\}, \text{ told, verb-crossing=TRUE}) \quad > \quad P(\text{NP}|\text{VP, VBD, } \{\}, \text{ told, verb-crossing=TRUE})$$

WH-MOVEMENT: A GPSG-STYLE TREATMENT



RESULTS

- Results on the Penn WSJ treebank
- Contribution of subcategorization, adjacency, verb-crossing
- Accuracy on different types of dependencies

Results on Section 23 of the Penn WSJ Treebank

MODEL	LR	LP
Magerman 95	84.0%	84.3%
Goodman 97	84.8%	85.3%
Collins 96	85.3%	85.7%
Charniak 97	86.7%	86.6%
Ratnaparkhi 97	86.3%	87.5%
Head-Driven Models	88.1%	88.3%

Also: Eisner 96 gives same dependency accuracy as Collins 96

LR = Labeled Recall
LP = Labeled Precision

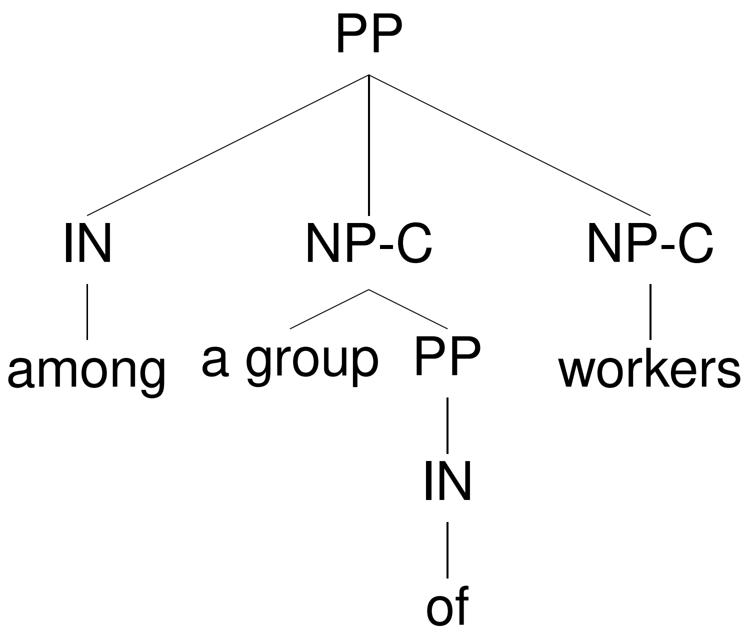
Contribution of Different Features

	LR	LP	
None	75.0%	76.5%	
Subcat	85.1%	86.8%	+10.2
Subcat + Adjacency	87.7%	87.8%	+1.8
Subcat + Adjacency + Verb	88.7%	89.0%	+1.1

	LR	LP	
None	75.0%	76.5%	
Adjacency	86.6%	86.7%	+10.9
Adjacency + Verb	87.8%	88.2%	+1.4
Adjacency + Verb + Subcat	88.7%	89.0%	+0.9

(Section 0 of the Penn WSJ Treebank)

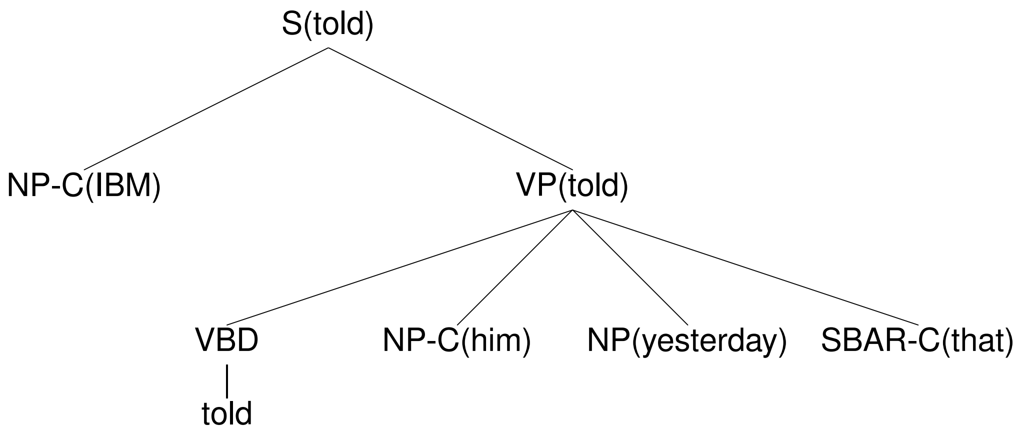
Subcategorization and Adjacency Overlap



Subcategorization and adjacency both fix this problem

EVALUATION OF DEPENDENCIES

- A sentence with n words has n dependencies



Head	Modifier	label	direction	description
told	IBM	`S VP NP-C`	Left	Subject
told	him	`VP TAG NP-C`	Right	Object
told	yesterday	`VP TAG NP`	Right	Adjunct
told	that	`VP TAG SBAR-C`	Right	SBAR complement

- Overall: 88.3% accuracy on section 0 (91% ignoring labels)

Type	Sub-type	Description	Count	Recall	Precision
Complement to a verb	S VP NP-C L	Subject	3248	95.75	95.11
	VP TAG NP-C R	Object	2095	92.41	92.15
6495 = 16.3% of all cases	VP TAG SBAR-C R		558	94.27	93.93
	...				
	TOTAL		6495	93.76	92.96
Other complements	PP TAG NP-C R		4335	94.72	94.04
	VP TAG VP-C R		1941	97.42	97.98
7473 = 18.8% of all cases	SBAR TAG S-C R		477	94.55	92.04
	...				
	TOTAL		7473	94.47	94.12
Mod'n within BaseNPs	NPB TAG TAG L		11786	94.60	93.46
	NPB TAG NPB L		358	97.49	92.82
12742 = 29.6% of all cases	NPB TAG TAG R		189	74.07	75.68
	...				
	TOTAL		12742	93.20	92.59
Sentential head	TOP TOP S R		1757	96.36	96.85
	TOP TOP SINV R		89	96.63	94.51
1917 = 4.8% of all cases	TOP TOP NP R		32	78.12	60.98
	TOP TOP SG R		15	40.00	33.33
	...				
	TOTAL		1917	94.99	94.99

Type	Sub-type	Description	Count	Recall	Precision
PP modification	NP NPB PP R		2112	84.99	84.35
	VP TAG PP R		1801	83.62	81.14
4473 = 11.2% of all cases	S VP PP L		287	90.24	81.96
	...				
	TOTAL		4473	82.29	81.51
Adjunct to a verb	VP TAG ADVP R		367	74.93	78.57
	VP TAG TAG R		349	90.54	93.49
2242 = 5.6% of all cases	VP TAG ADJP R		259	83.78	80.37
	...				
	TOTAL		2242	75.11	78.44
Mod'n to NPs	NP NPB NP R	Appositive	495	74.34	75.72
	NP NPB SBAR R	Relative clause	476	79.20	79.54
1418 = 3.6% of all cases	NP NPB VP R	Reduced relative	205	77.56	72.60
	...				
	TOTAL		1418	73.20	75.49
Coordination	NP NP NP R		289	55.71	53.31
	VP VP VP R		174	74.14	72.47
763 = 1.9% of all cases	S S S R		129	72.09	69.92
	...				
	TOTAL		763	61.47	62.20

Some Thoughts about Related Work

- SPATTER: the importance of the choice of decomposition
- Charniak 97: the importance of breaking down rules

SPATTER (Magerman 95, Jelinek et. al 94)

Representation Context-free trees with head-words

Decomposition d_i is the i'th decision in a left-to-right, bottom-up parse of the tree

$$P(T|S) = \prod_{i=1\ldots n} P(d_i | d_1 \ldots d_{i-1}, S)$$

Independence Assumptions $\phi(d_1 \ldots d_{i-1})$ is found automatically using decision trees

Problems with SPATTER

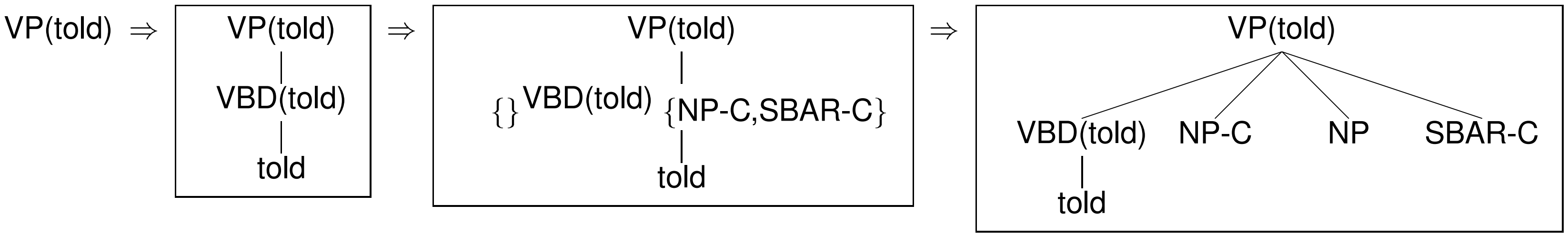
VB	NP	P	NP		
VB	P	NP	P	NP	
VB	ADVP	P	NP	P	NP

PROBLEMS WITH SPATTER

N	V	N	CC	N	V	N
John	likes	Mary	and	Bill	loves	Jill

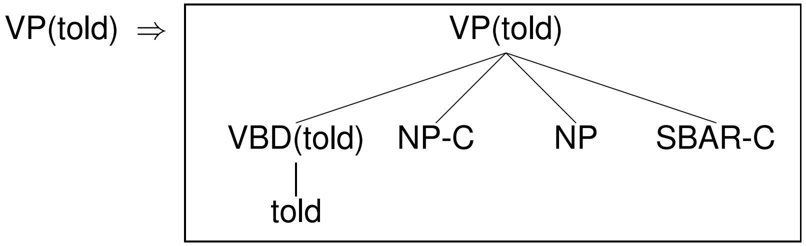
A CONTRAST WITH CHARNIAK 97

- Generation of a rule is broken down into smaller steps



- The model can generalize to produce rules in test data that have not been seen in training

- Charniak 97: entire rule is expanded in one step



THE PENN TREEBANK HAS MANY RULES

- 17.1% of sentences in test data have a rule not seen in training

Chomsky Adjunction	Penn Treebank
VP $\rightarrow$ V NP-C	VP $\rightarrow$ V NP-C
VP $\rightarrow$ VP PP	VP $\rightarrow$ V NP-C PP
	VP $\rightarrow$ V NP-C PP
	VP $\rightarrow$ V NP-C PP PP
	VP $\rightarrow$ V NP-C PP PP PP ...

- With good motivation: VP $\rightarrow$ NP-C NP SBAR-C

THE IMPACT OF COVERAGE ON ACCURACY

MODEL	LR	LP	CBs	0 CBs	$\leq$ 2 CBs
Full model	88.8	89.0	0.94	65.9	85.6
Full model (restricted)	87.9	87.0	1.19	62.5	82.4

Future Work: Improving Accuracy

- Improving accuracy:
  - Increased Context/Improved Estimation
  - Unsupervised Learning
- Deeper Analysis:
  - Non-constituent coordination, wh-movement of phrases other than NPs, PRO-control, tough raising etc. etc.
  - Mapping to theta roles
  - General information extraction from parse trees

FUTURE WORK: OTHER LANGUAGES

- Old/Middle English
- Czech. 1998 Johns Hopkins Summer Workshop:
  - 82% dependency accuracy
  - Major problem is inflection. Need parameters

$$P(\text{modifier tag}|\text{head tag})$$

$$P(\text{word form}|\text{word stem, tag})$$

SUMMARY

- What to count? **Lexically conditioned parameters:**
  - Head-projection
  - Subcategorization
  - Placement of complements/adjuncts
  - Dependencies
  - Close-attachment/Wh-movement

- How to combine the counts? **History-based Approach:**
  - Representation = Lexicalized trees
  - Decomposition = head-centered, top-down derivation
- **Results:**
  - Over 88% constituent accuracy
  - Over 90% accuracy on dependencies

A FINAL POINT

- Prior knowledge is unavoidable:
  - History-based models generalize practically all parsing models
  - The choice of **decomposition** is crucial, implies a substantial **bias**
  - Prior linguistic knowledge is embedded in the choice of decomposition
  - Decomposition should be motivated by concerns about **locality**
- The learning component shouldn't be underestimated:
  - Volume of information: 780,000 dependency events (390,000 distinct dependency types), over 9,000,000 dependency counts
  - Blends many different knowledge sources into a consistent model (subcategorization, dependencies, close-attachment etc.)
  - Balances fine-grained lexical statistics against coarser statistics (backed-off estimation)