

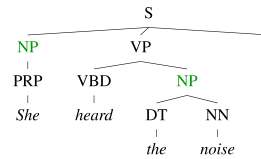
# Statistical NLP Spring 2008



## Lecture 16: PCFGs

Dan Klein – UC Berkeley

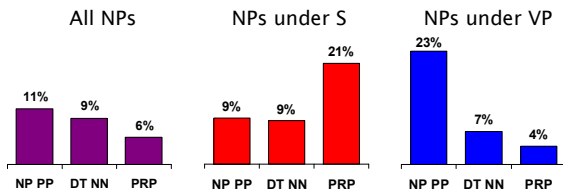
# Context Freedom?



- Not every NP expansion can fill every NP slot
  - A grammar with symbols like "NP" won't be context-free
  - Statistically, treebank symbols represent overly strong conditional independence assumptions

# Non-Independence I

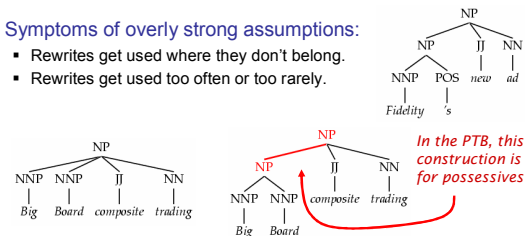
- Independence assumptions are often too strong.



- Example: the expansion of an NP is highly dependent on the parent of the NP (i.e., subjects vs. objects).
- Also: the subject and object expansions are correlated!

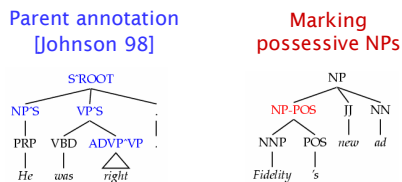
# Non-Independence II

- Who cares?
  - NB, HMMs, all make false assumptions!
  - For **generation**, consequences would be obvious.
  - For **parsing**, does it impact accuracy?
- Symptoms of overly strong assumptions:
  - Rewrites get used where they don't belong.
  - Rewrites get used too often or too rarely.



# Breaking Up the Symbols

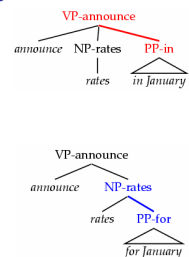
- We can relax independence assumptions by encoding dependencies into the PCFG symbols:



- What are the most useful "features" to encode?

# Lexicalization

- Lexical heads important for certain classes of ambiguities (e.g., PP attachment):
- Lexicalizing grammar creates a much larger grammar. (cf. next week)
  - Sophisticated smoothing needed
  - Smarter parsing algorithms
  - More data needed
- How necessary is lexicalization?
  - Bilexical vs. monolexical selection
  - Closed vs. open class lexicalization



## Typical Experimental Setup

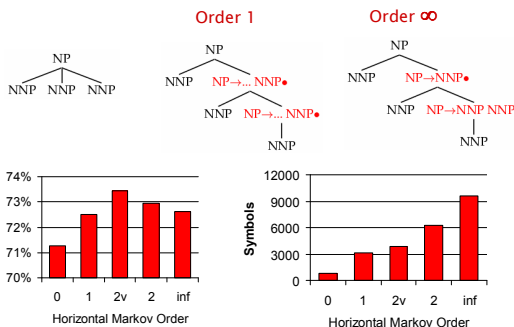
- Corpus: Penn Treebank, WSJ



Training: sections 02-21  
 Development: section 22 (here, first 20 files)  
 Test: section 23

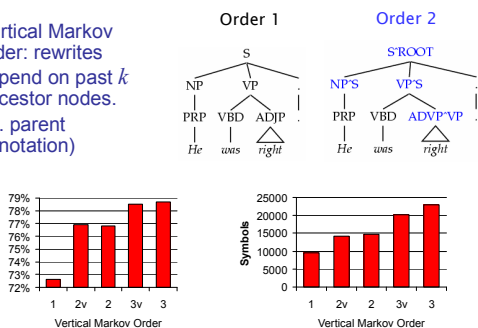
- Accuracy – F1: harmonic mean of per-node labeled precision and recall.
- Here: also size – number of symbols in grammar.
  - Passive / complete symbols: NP, NP^S
  - Active / incomplete symbols: NP → NP CC •

## Horizontal Markovization

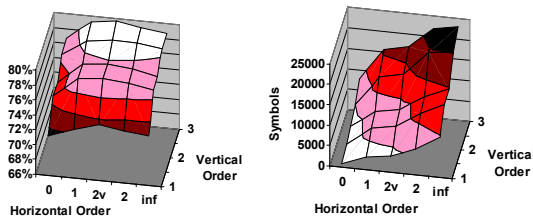


## Vertical Markovization

- Vertical Markov order: rewrites depend on past  $k$  ancestor nodes. (cf. parent annotation)



## Vertical and Horizontal



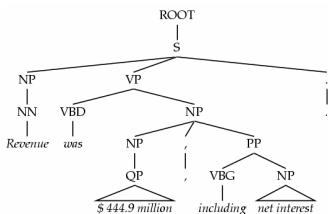
- Examples:

- Raw treebank:  $v=1, h=\infty$
- Johnson 98:  $v=2, h=\infty$
- Collins 99:  $v=2, h=2$
- Best F1:  $v=3, h=2v$

Model	F1	Size
Base: $v=h=2v$	77.8	7.5K

## Unary Splits

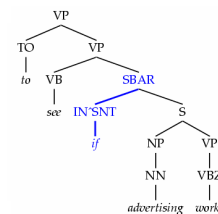
- Problem: unary rewrites used to transmute categories so a high-probability rule can be used.
- Solution: Mark unary rewrite sites with -U



Annotation	F1	Size
Base	77.8	7.5K
UNARY	78.3	8.0K

## Tag Splits

- Problem: Treebank tags are too coarse.
- Example: Sentential, PP, and other prepositions are all marked IN.
- Partial Solution:
  - Subdivide the IN tag.



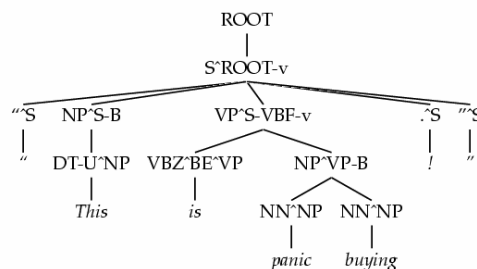
Annotation	F1	Size
Previous	78.3	8.0K
SPLIT-IN	80.3	8.1K

## Other Tag Splits

- UNARY-DT: mark demonstratives as DT^U ("the X" vs. "those")
- UNARY-RB: mark phrasal adverbs as RB^U ("quickly" vs. "very")
- TAG-PA: mark tags with non-canonical parents ("not" is an RB^VP)
- SPLIT-AUX: mark auxiliary verbs with -AUX [cf. Charniak 97]
- SPLIT-CC: separate "but" and "&" from other conjunctions
- SPLIT-%: "%" gets its own tag.

	F1	Size
UNARY-DT	80.4	8.1K
UNARY-RB	80.5	8.1K
TAG-PA	81.2	8.5K
SPLIT-AUX	81.6	9.0K
SPLIT-CC	81.7	9.1K
SPLIT-%	81.8	9.3K

## A Fully Annotated (Unlex) Tree

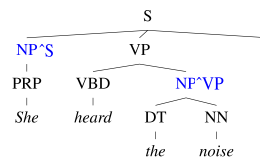


## Some Test Set Results

Parser	LP	LR	F1	CB	0 CB
Magerman 95	84.9	84.6	84.7	1.26	56.6
Collins 96	86.3	85.8	86.0	1.14	59.9
Unlexicalized	86.9	85.7	86.3	1.10	60.3
Charniak 97	87.4	87.5	87.4	1.00	62.1
Collins 99	88.7	88.6	88.6	0.90	67.1

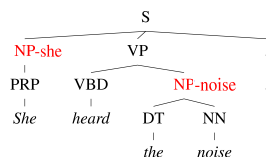
- Beats "first generation" lexicalized parsers.
- Lots of room to improve – more complex models next.

## The Game of Designing a Grammar



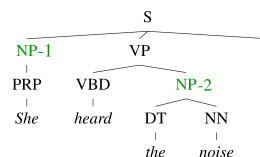
- Annotation refines base treebank symbols to improve statistical fit of the grammar
  - Parent annotation [Johnson '98]

## The Game of Designing a Grammar



- Annotation refines base treebank symbols to improve statistical fit of the grammar
  - Parent annotation [Johnson '98]
  - Head lexicalization [Collins '99, Charniak '00]

## The Game of Designing a Grammar



- Annotation refines base treebank symbols to improve statistical fit of the grammar
  - Parent annotation [Johnson '98]
  - Head lexicalization [Collins '99, Charniak '00]
  - Automatic clustering?

## Manual Annotation

### Manually split categories

- NP: subject vs object
- DT: determiners vs demonstratives
- IN: sentential vs prepositional

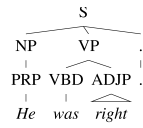
### Advantages:

- Fairly compact grammar
- Linguistic motivations

### Disadvantages:

- Performance leveled out
- Manually annotated

Model	F1
Naïve Treebank Grammar	72.6
Klein & Manning '03	86.3



## Automatic Annotation Induction

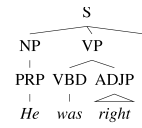
### Advantages:

- Automatically learned: Label all nodes with latent variables. Same number  $k$  of subcategories for all categories.

### Disadvantages:

- Grammar gets too large
- Most categories are oversplit while others are undersplit.

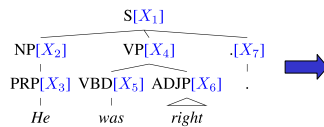
Model	F1
Klein & Manning '03	86.3
Matsuzaki et al. '05	86.7



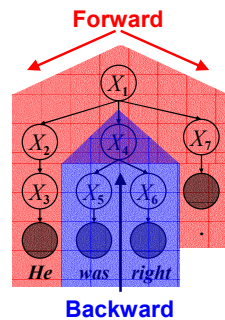
## Learning Latent Annotations

### EM algorithm:

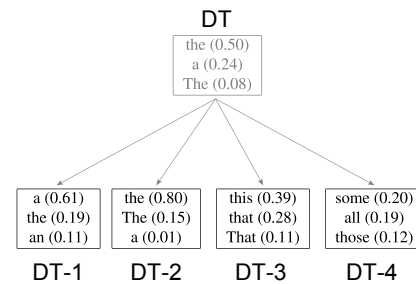
- Brackets are known
- Base categories are known
- Only induce subcategories



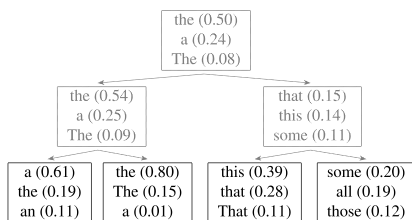
Just like Forward-Backward for HMMs.



## Refinement of the DT tag

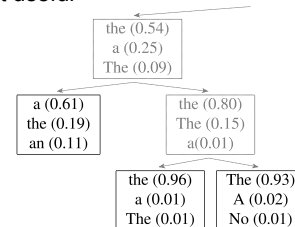


## Hierarchical refinement



## Adaptive Splitting

- Want to split complex categories more
- Idea: split everything, roll back splits which were least useful





## Learned Splits

- Proper Nouns (NNP):

NNP-14	Oct.	Nov.	Sept.
NNP-12	John	Robert	James
NNP-2	J.	E.	L.
NNP-1	Bush	Noriega	Peters
NNP-15	New	San	Wall
NNP-3	York	Francisco	Street

- Personal pronouns (PRP):

PRP-0	it	He	I
PRP-1	it	he	they
PRP-2	it	them	him

## Learned Splits

- Relative adverbs (RBR):

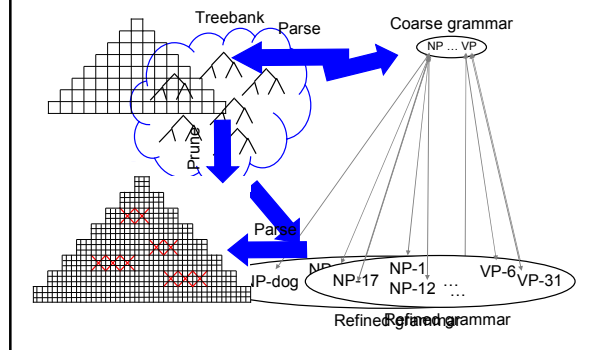
RBR-0	further	lower	higher
RBR-1	more	less	More
RBR-2	earlier	Earlier	later

- Cardinal Numbers (CD):

CD-7	one	two	Three
CD-4	1989	1990	1988
CD-11	million	billion	trillion
CD-0	1	50	100
CD-3	1	30	31
CD-9	78	58	34

## Coarse-to-Fine Parsing

[Goodman '97, Charniak&Johnson '05]

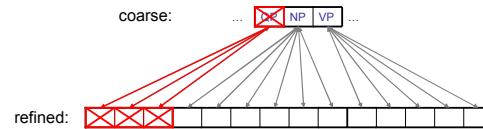


## Prune?

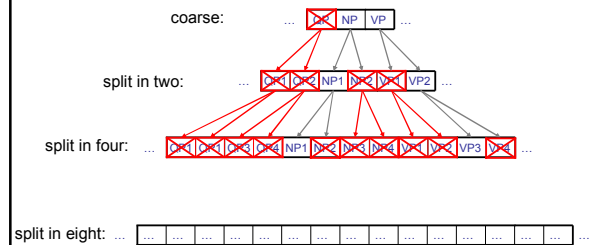
For each chart item  $X[i, j]$ , compute posterior probability:

$$\frac{P_{IN}(X, i, j) \cdot P_{OUT}(X, i, j)}{P_{IN}(root, 0, n)} < \text{threshold}$$

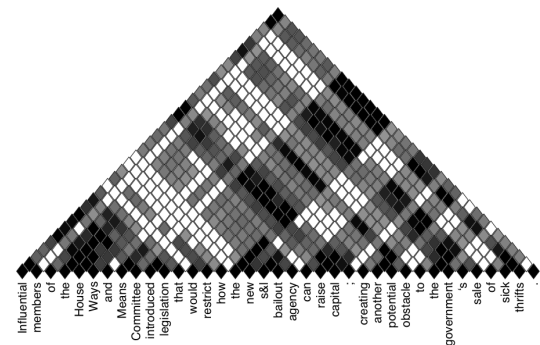
E.g. consider the span 5 to 12:



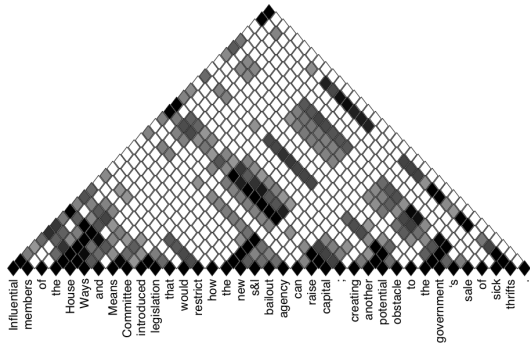
## Hierarchical Pruning



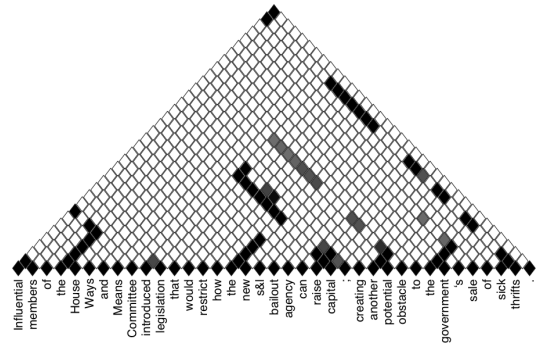
## Bracket Posteriors (after $G_0$ )



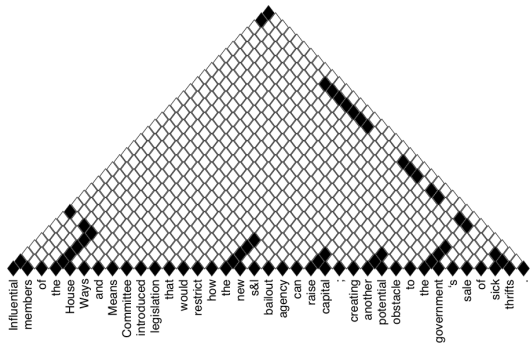
## Bracket Posteriors (after $G_1$ )



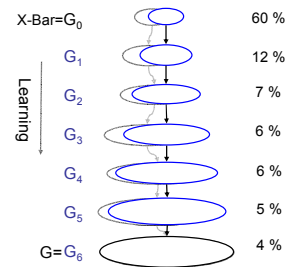
## Bracket Posteriors (Final)



## Bracket Posteriors (Best Tree)



## Parsing times



**1621 min**  
**111 min**  
**35 min**  
**15 min**  
 (no search error)