# Statistical NLP

## Spring 2010

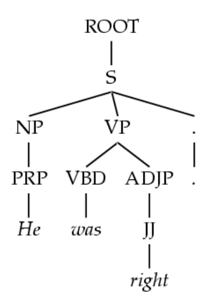
## Lecture 14: PCFGs

Dan Klein – UC Berkeley

# Treebank PCFGs

[Charniak 96]

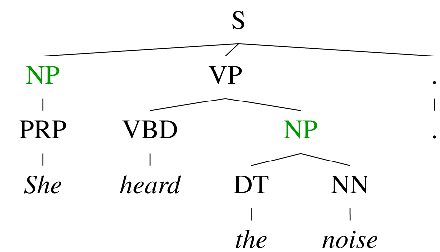
- Use PCFGs for broad coverage parsing
- Can take a grammar right off the trees (doesn't work well):




| | |
|---|---|
| ROOT → S | 1 |
| S → NP VP . | 1 |
| NP → PRP | 1 |
| VP → VBD ADJP | 1 |
| ….. | |

| *Model* | *F1* |
|---|---|
| Baseline | 72.0 |

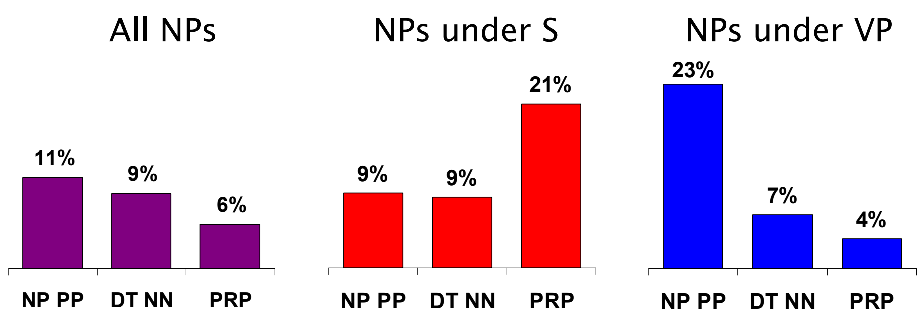
# Conditional Independence?




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

# Non-Independence

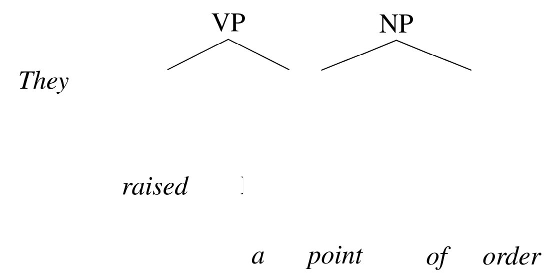
- Independence assumptions are often too strong.



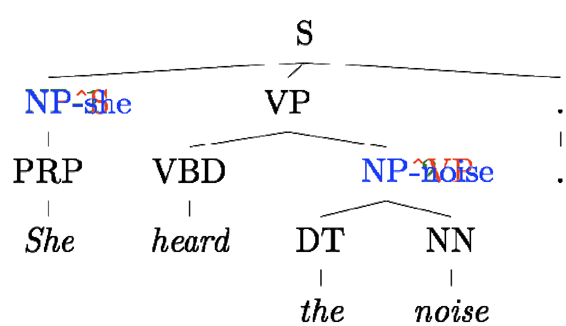

- Example: the expansion of an NP is highly dependent on the parent of the NP (i.e., subjects vs. objects).
- Also: the subject and object expansions are correlated!

## Grammar Refinement

- Example: PP attachment

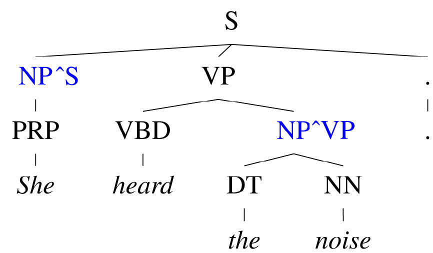



## Grammar Refinement




- Structure Annotation [Johnson ’98, Klein&Manning ’03]
- Lexicalization [Collins ’99, Charniak ’00]
- Latent Variables [Matsuzaki et al. 05, Petrov et al. ’06]

# The Game of Designing a Grammar




- Annotation refines base treebank symbols to improve statistical fit of the grammar
  - Structural annotation

# Typical Experimental Setup

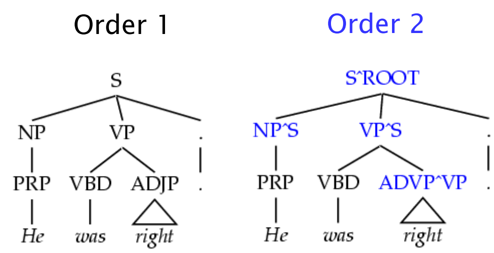
- Corpus: Penn Treebank, WSJ

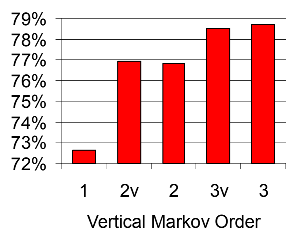
| | | |
|---|---|---|
| Training: | sections | 02-21 |
| Development: | section | 22 (here, first 20 files) |
| Test: | section | 23 |

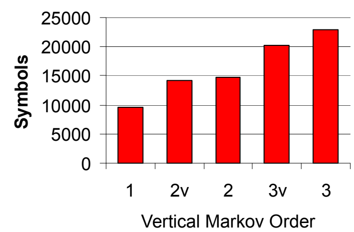
- Accuracy – F1: harmonic mean of per-node labeled precision and recall.
- Here: also size – number of symbols in grammar.
  - Passive / complete symbols: NP, NP^S
  - Active / incomplete symbols: NP → NP CC •

# Vertical Markovization

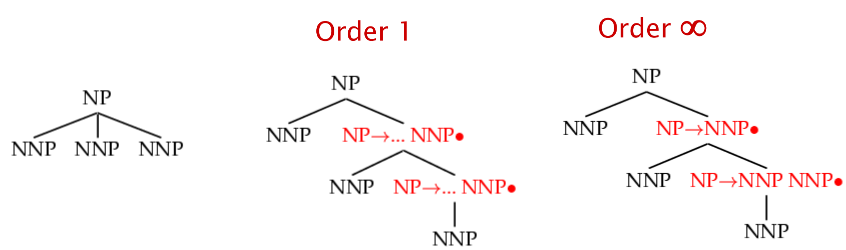
- Vertical Markov order: rewrites depend on past $k$ ancestor nodes.
  (cf. parent annotation)

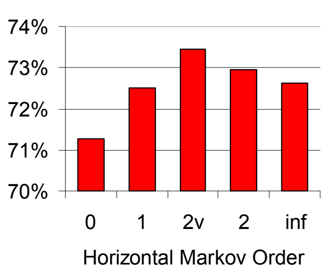


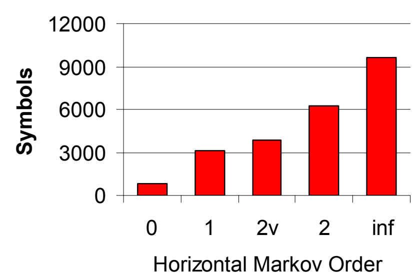




# Horizontal Markovization

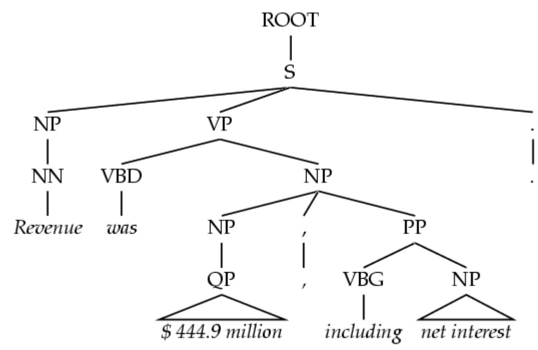


## Unary Splits

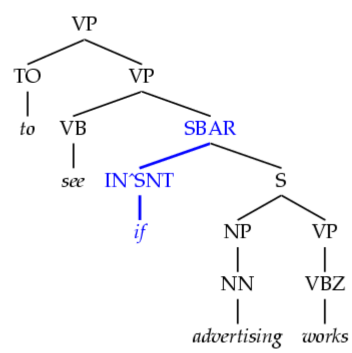
- Problem: unary rewrites used to transmute categories so a high-probability rule can be used.
- Solution: Mark unary rewrite sites with -U




| 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.



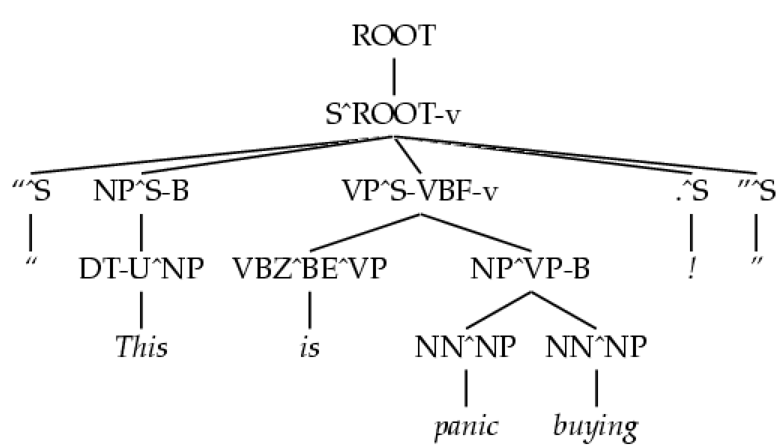

| Annotation | F1 | Size |
|---|---|---|
| Previous | 78.3 | 8.0K |
| SPLIT-IN | 80.3 | 8.1K |

# 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 |
|---|---|
| 80.4 | 8.1K |
| 80.5 | 8.1K |
| 81.2 | 8.5K |
| 81.6 | 9.0K |
| 81.7 | 9.1K |
| 81.8 | 9.3K |

# A Fully Annotated (Unlex) Tree

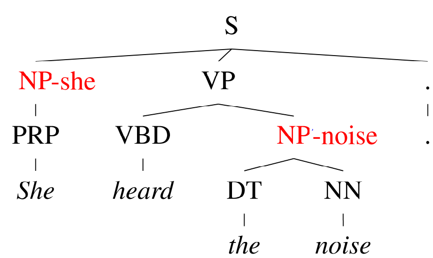


## Some Test Set Results

| 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 |

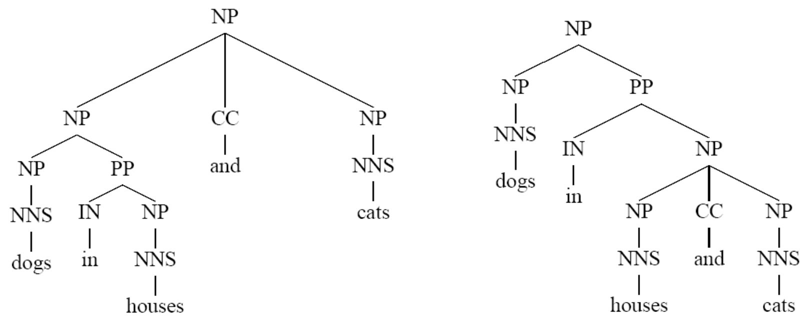
- Beats “first generation” lexicalized parsers.
- Lots of room to improve – more complex models next.

## The Game of Designing a Grammar




- Annotation refines base treebank symbols to improve statistical fit of the grammar
  - Structural annotation [Johnson ’98, Klein and Manning 03]
  - Head lexicalization [Collins ’99, Charniak ’00]

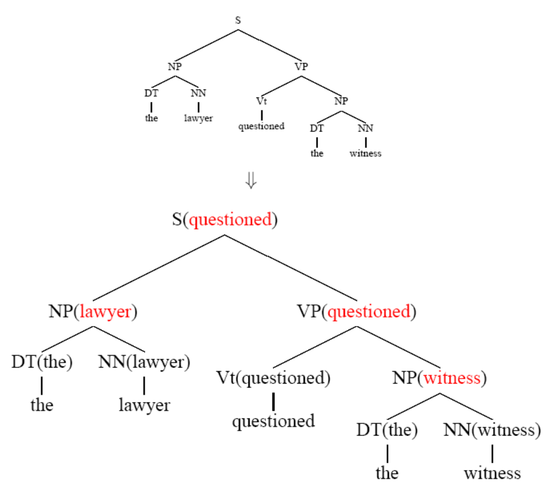
# Problems with PCFGs




- What's different between basic PCFG scores here?
- What (lexical) correlations need to be scored?

# Lexicalized Trees

- Add "headwords" to each phrasal node
  - Syntactic vs. semantic heads
  - Headship not in (most) treebanks
  - Usually *use head rules*, e.g.:
    - NP:
      - Take leftmost NP
      - Take rightmost N*
      - Take rightmost JJ
      - Take right child
    - VP:
      - Take leftmost VB*
      - Take leftmost VP
      - Take left child

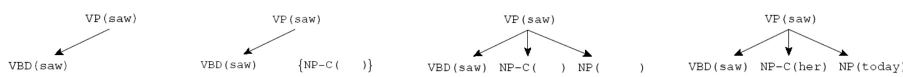


# Lexicalized PCFGs?

- Problem: we now have to estimate probabilities like

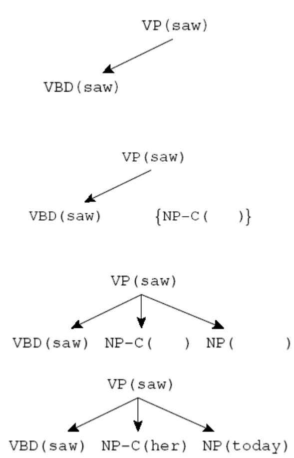
```
VP(saw) -> VBD(saw) NP-C(her) NP(today)
```

- Never going to get these atomically off of a treebank
- Solution: break up derivation into smaller steps




# Lexical Derivation Steps

- A derivation of a local tree [Collins 99]



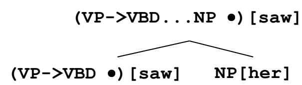

Choose a head tag and word

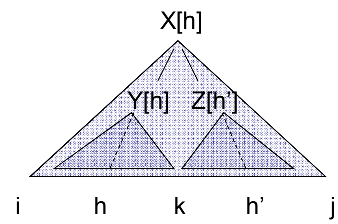
Choose a complement bag

Generate children (incl. adjuncts)

Recursively derive children

# Lexicalized CKY

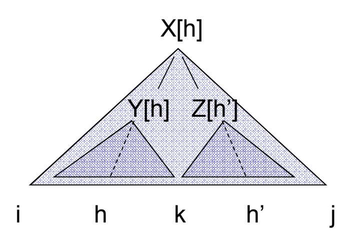



```
bestScore(X,i,j,h)
  if (j = i+1)
    return tagScore(X,s[i])
  else
    return
      max max            score(X[h]->Y[h] Z[h']) *
        k,h',X->YZ
                         bestScore(Y,i,k,h) *
                         bestScore(Z,k,j,h')
          max            score(X[h]->Y[h'] Z[h]) *
        k,h',X->YZ
                         bestScore(Y,i,k,h') *
                         bestScore(Z,k,j,h)
```

# Pruning with Beams

- The Collins parser prunes with per-cell beams [Collins 99]
  - Essentially, run the $O(n^5)$ CKY
  - Remember only a few hypotheses for each span <i,j>.
  - If we keep K hypotheses at each span, then we do at most $O(nK^2)$ work per span (why?)
  - Keeps things more or less cubic
- Also: certain spans are forbidden entirely on the basis of punctuation (crucial for speed)



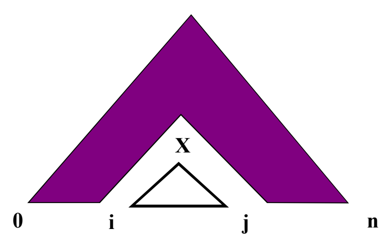
## Pruning with a PCFG

- The Charniak parser prunes using a two-pass approach [Charniak 97+]
  - First, parse with the base grammar
  - For each X:[i,j] calculate P(X|i,j,s)
    - This isn't trivial, and there are clever speed ups
  - Second, do the full $O(n^5)$ CKY
    - Skip any X :[i,j] which had low (say, < 0.0001) posterior
  - Avoids almost all work in the second phase!

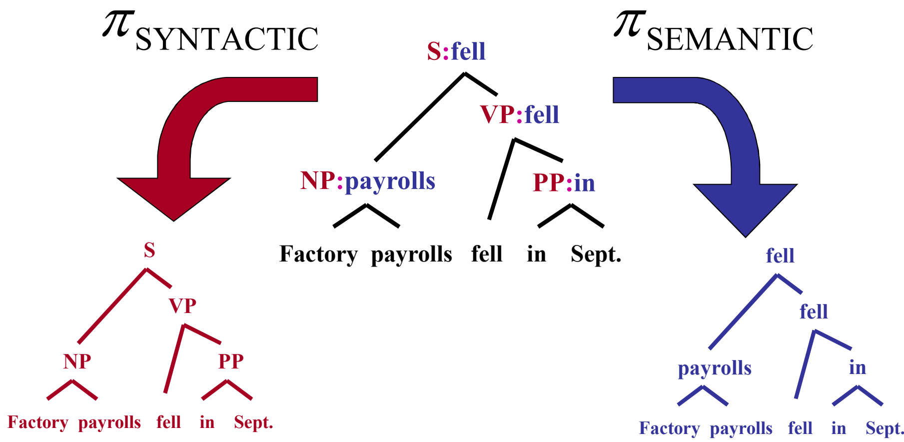
- Charniak et al 06: can use more passes
- Petrov et al 07: can use many more passes

## Pruning with A*

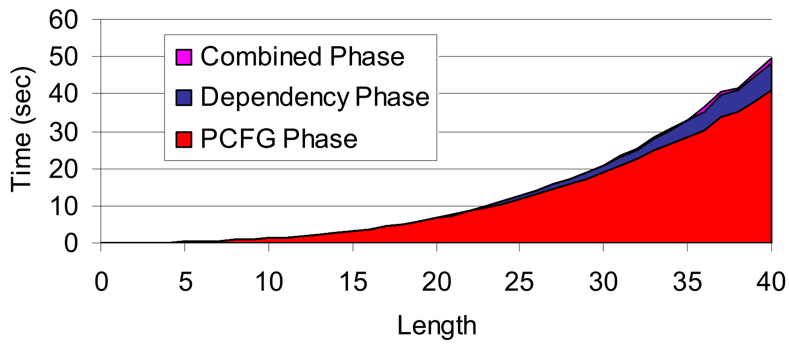
- You can also speed up the search without sacrificing optimality
- For agenda-based parsers:
  - Can select which items to process first
  - Can do with any "figure of merit" [Charniak 98]
  - If your figure-of-merit is a valid A* heuristic, no loss of optimiality [Klein and Manning 03]



## Projection-Based A*




## A* Speedup



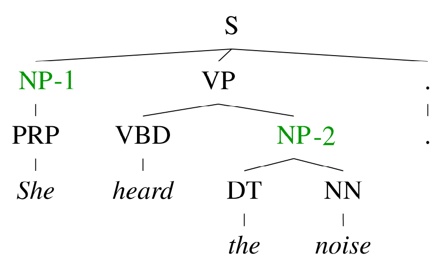

- Total time dominated by calculation of A* tables in each projection… $O(n^3)$

# Results

- Some results
  - Collins 99 – 88.6 F1 (generative lexical)
  - Charniak and Johnson 05 – 89.7 / 91.3 F1 (generative lexical / reranked)
  - Petrov et al 06 – 90.7 F1 (generative unlexical)
  - McClosky et al 06 – 92.1 F1 (gen + rerank + self-train)

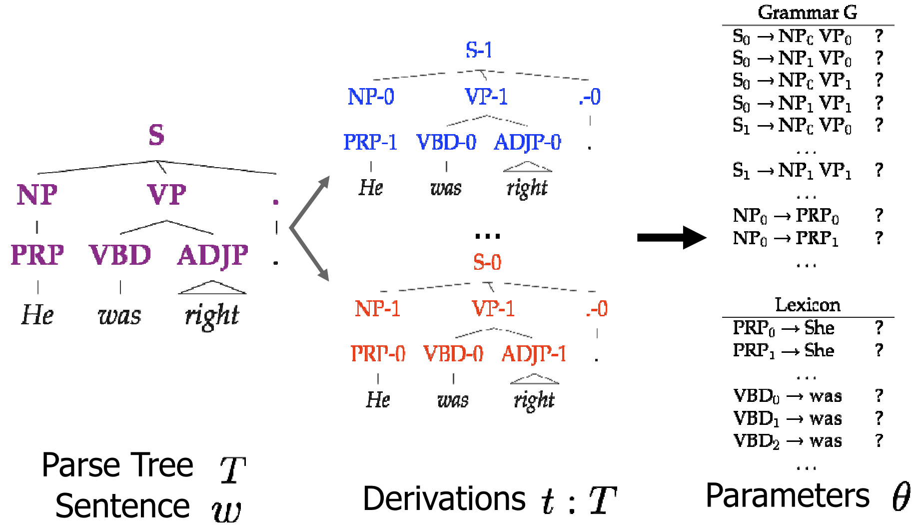
- However
  - Bilexical counts rarely make a difference (why?)
  - Gildea 01 – Removing bilexical counts costs < 0.5 F1

# The Game of Designing a Grammar




- Annotation refines base treebank symbols to improve statistical fit of the grammar
  - Structural annotation
  - Head lexicalization
  - Automatic clustering?

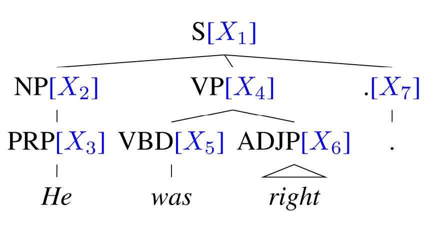
# Latent Variable Grammars

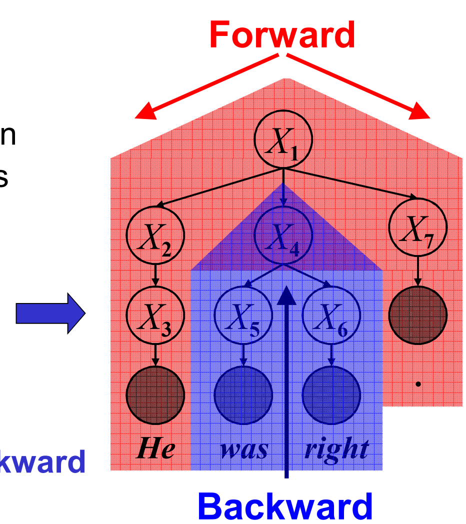



# Learning Latent Annotations

EM algorithm:

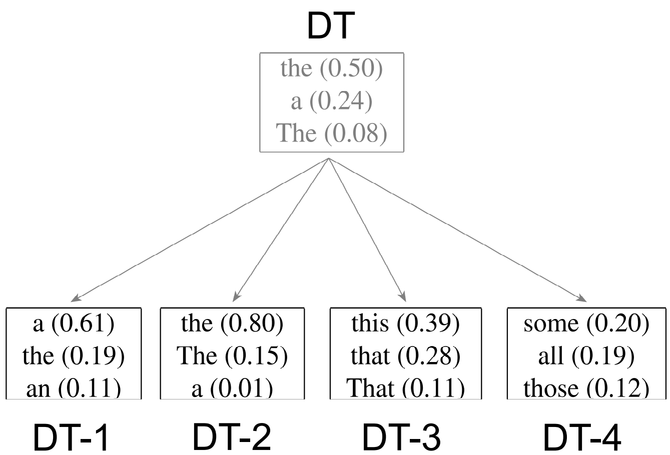
- Brackets are known
- Base categories are known
- Only induce subcategories



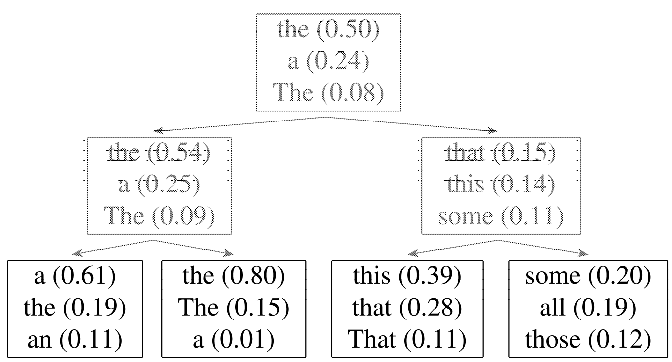




**Just like Forward-Backward for HMMs.**

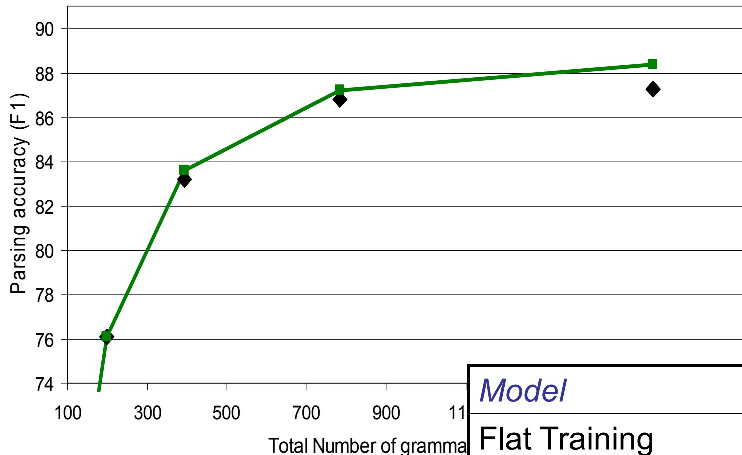
# Refinement of the DT tag




# Hierarchical refinement



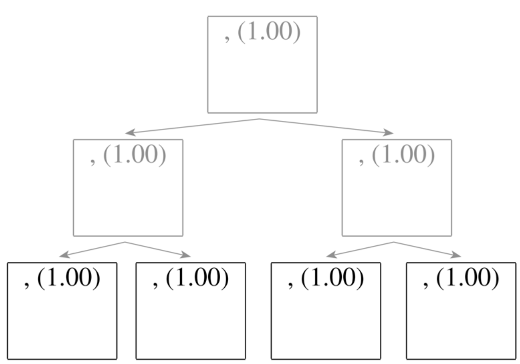
# Hierarchical Estimation Results




| *Model* | *F1* |
|---|---|
| Flat Training | 87.3 |
| Hierarchical Training | 88.4 |

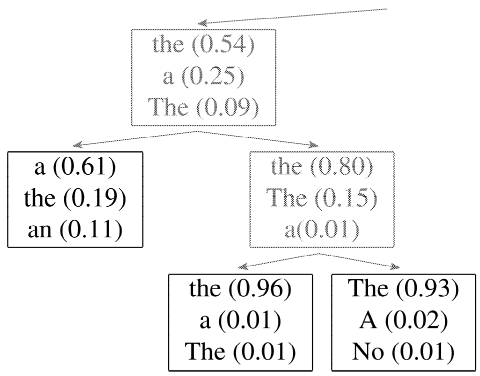
# Refinement of the , tag

- Splitting all categories equally is wasteful:

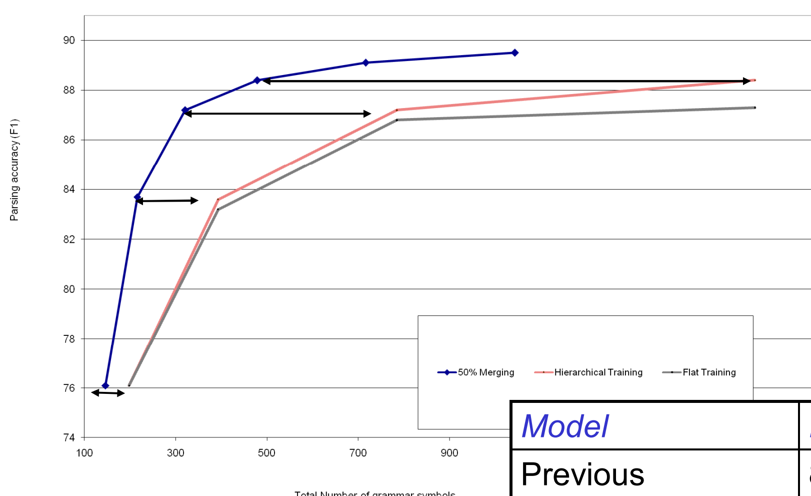


# Adaptive Splitting

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

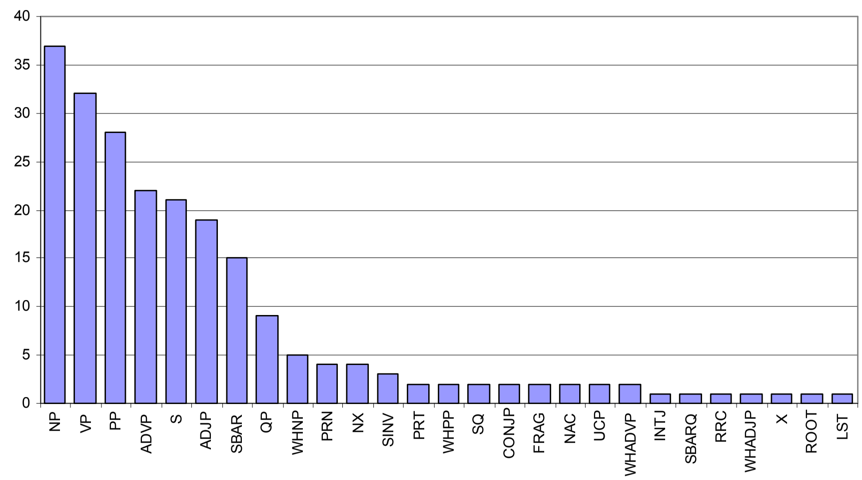



# Adaptive Splitting Results

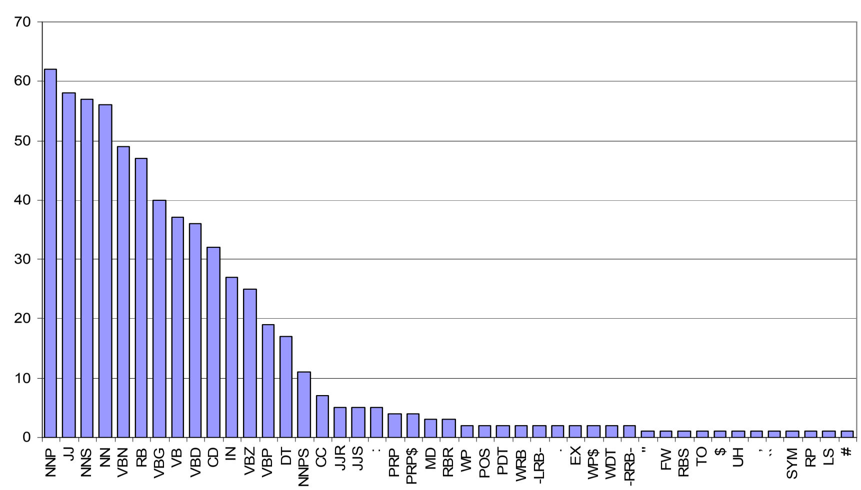



| Model | F1 |
|---|---|
| Previous | 88.4 |
| With 50% Merging | 89.5 |

# Number of Phrasal Subcategories




# Number of Lexical Subcategories



## 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):

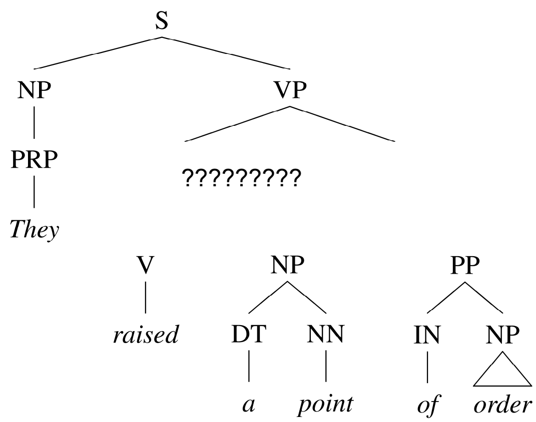
| | | | |
|---|---|---|---|
| RBR-0 | further | lower | higher |
| RBR-1 | more | less | More |
| RBR-2 | earlier | Earlier | later |

- 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 Inference

- Example: PP attachment

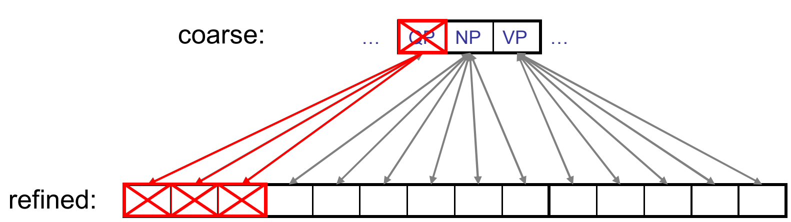



# Prune?

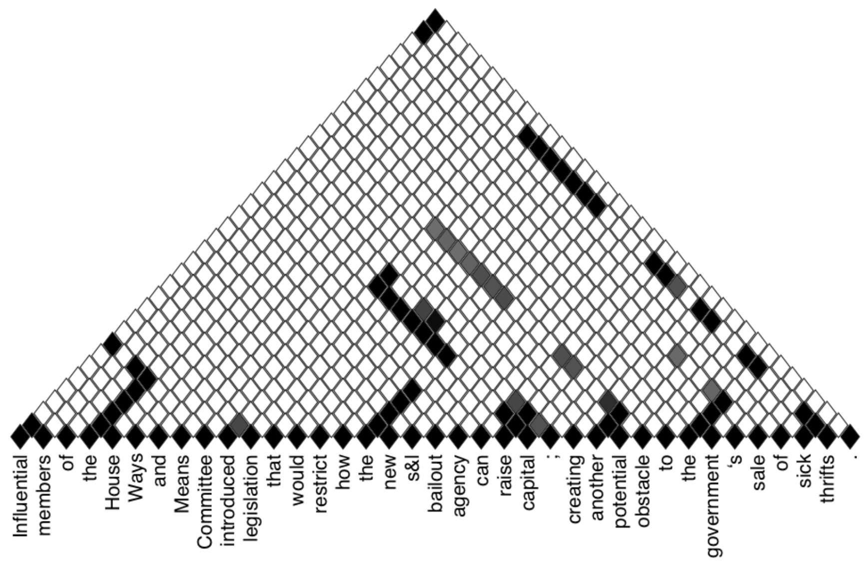
For each chart item $X[i,j]$, compute posterior probability:

$$\frac{\mathrm{P}_{\mathrm{IN}}(X,i,j)\cdot \mathrm{P}_{\mathrm{OUT}}(X,i,j)}{\mathrm{P}_{\mathrm{IN}}(root,0,n)} < \textit{threshold}$$

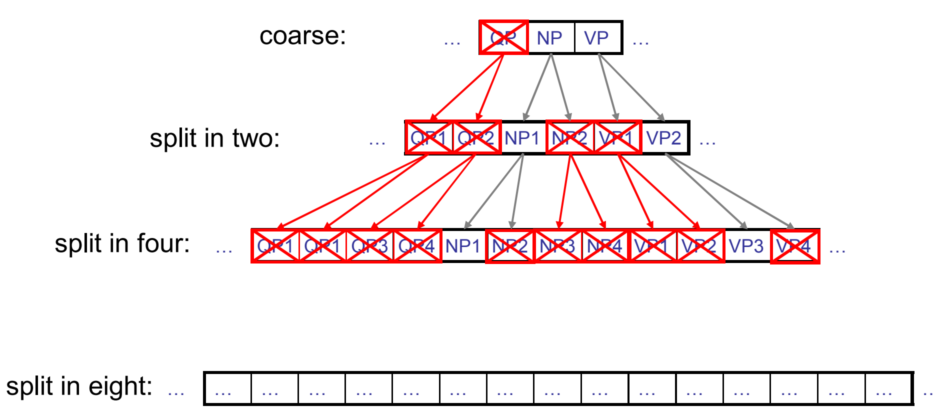
E.g. consider the span 5 to 12:



# Bracket Posteriors




# Hierarchical Pruning



# Final Results (Accuracy)

| | | ≤ 40 words F1 | all F1 |
|---|---|---|---|
| ENG | Charniak&Johnson '05 (generative) | 90.1 | 89.6 |
| | **Split / Merge** | **90.6** | **90.1** |
| GER | Dubey '05 | 76.3 | - |
| | **Split / Merge** | **80.8** | **80.1** |
| CHN | Chiang et al. '02 | 80.0 | 76.6 |
| | **Split / Merge** | **86.3** | **83.4** |

Still higher numbers from reranking / self-training methods