Parser Evaluation

Parser Evaluation

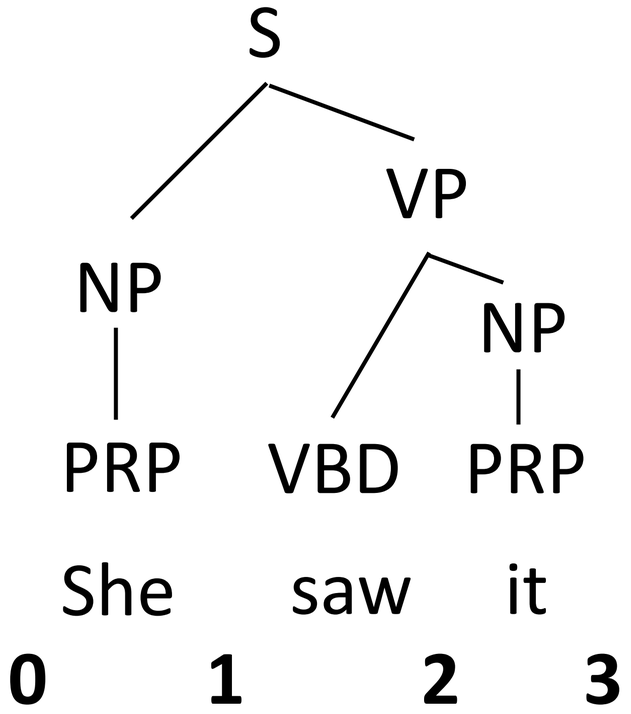
- View a parse as a set of labeled *brackets* / constituents

S(0,3)

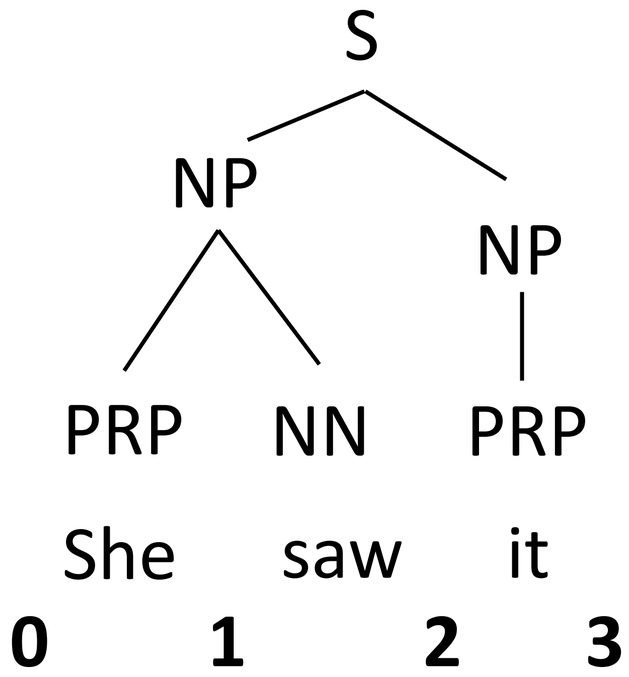
NP(0,1)

PRP(0,1) (but standard evaluation *does not count POS tags*)

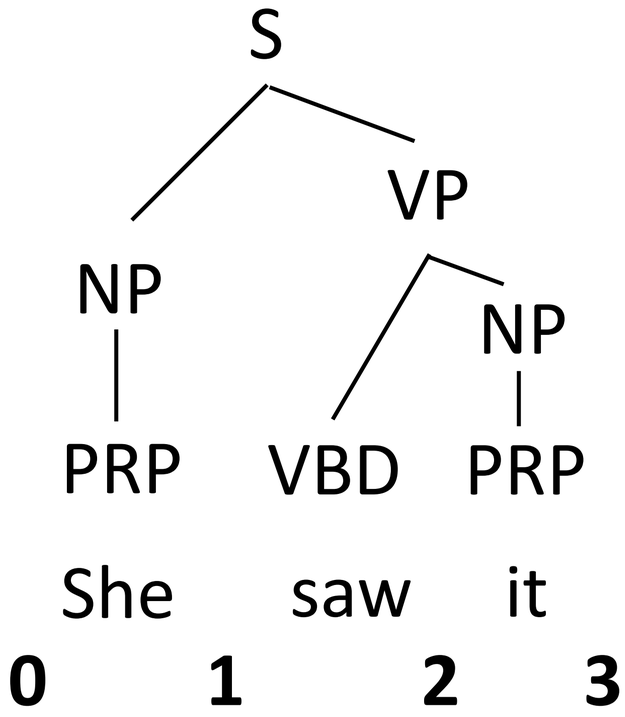
VP(1,3), VBD(1,2), NP(2,3), PRP(2,3)



Parser Evaluation



S(0,3),
NP(0,2),
NP(2,3),
~~PRP(0,1)~~,
~~NN(1,2)~~,
~~PRP(2,3)~~



S(0,3),
NP(0,1),
VP(1,3),
NP(2,3),
~~PRP(0,1)~~,
~~VBD(1,2)~~,
~~PRP(2,3)~~

- Precision: number of correct predictions / number of predictions = 2/3
- Recall: number of correct predictions / number of golds = 2/4
- F1: harmonic mean of precision and recall = $(1/2 * ((2/4)^{-1} + (2/3)^{-1}))^{-1}$

  $= 0.57$ (closer to min)

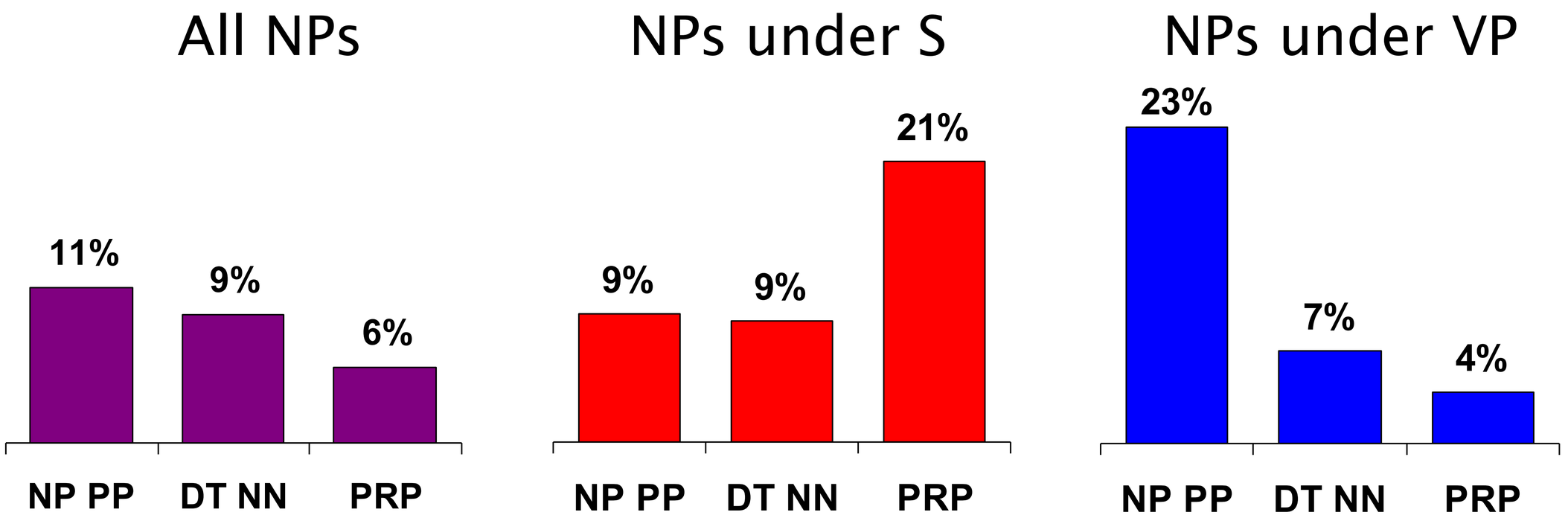
Results

- Standard dataset for English: Penn Treebank (Marcus et al., 1993)
- “Vanilla” PCFG: ~71 F1
- Best PCFGs for English: ~90 F1
- State-of-the-art discriminative models (using unlabeled data): 95 F1
- Other languages: results vary widely depending on annotation + complexity of the grammar

Klein and Manning (2003)

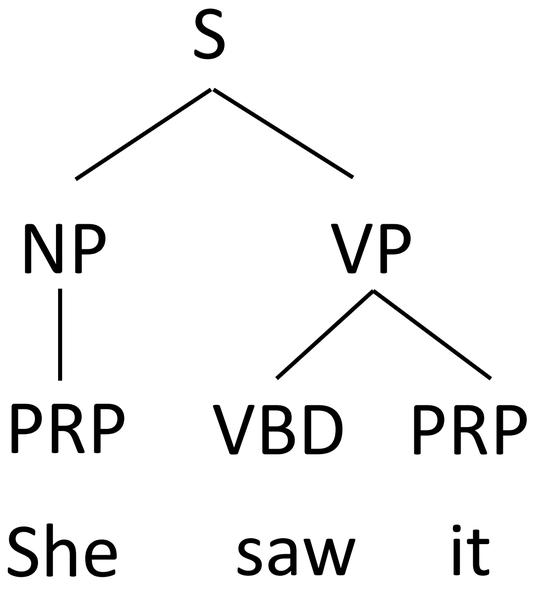
Refining Generative Grammars

PCFG Independence Assumptions

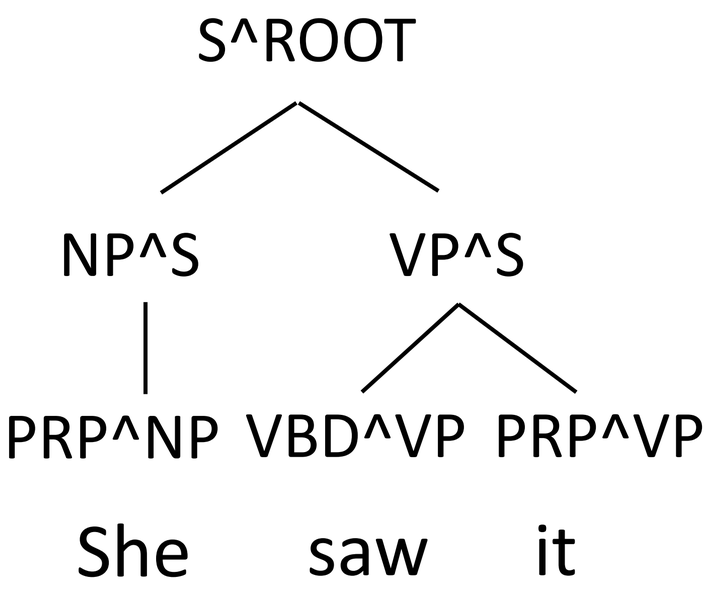


- Language is not context-free: NPs in different contexts rewrite differently
- [They]$_{NP}$ received [the package of books]$_{NP}$

Vertical Markovization



Basic tree (v = 1)

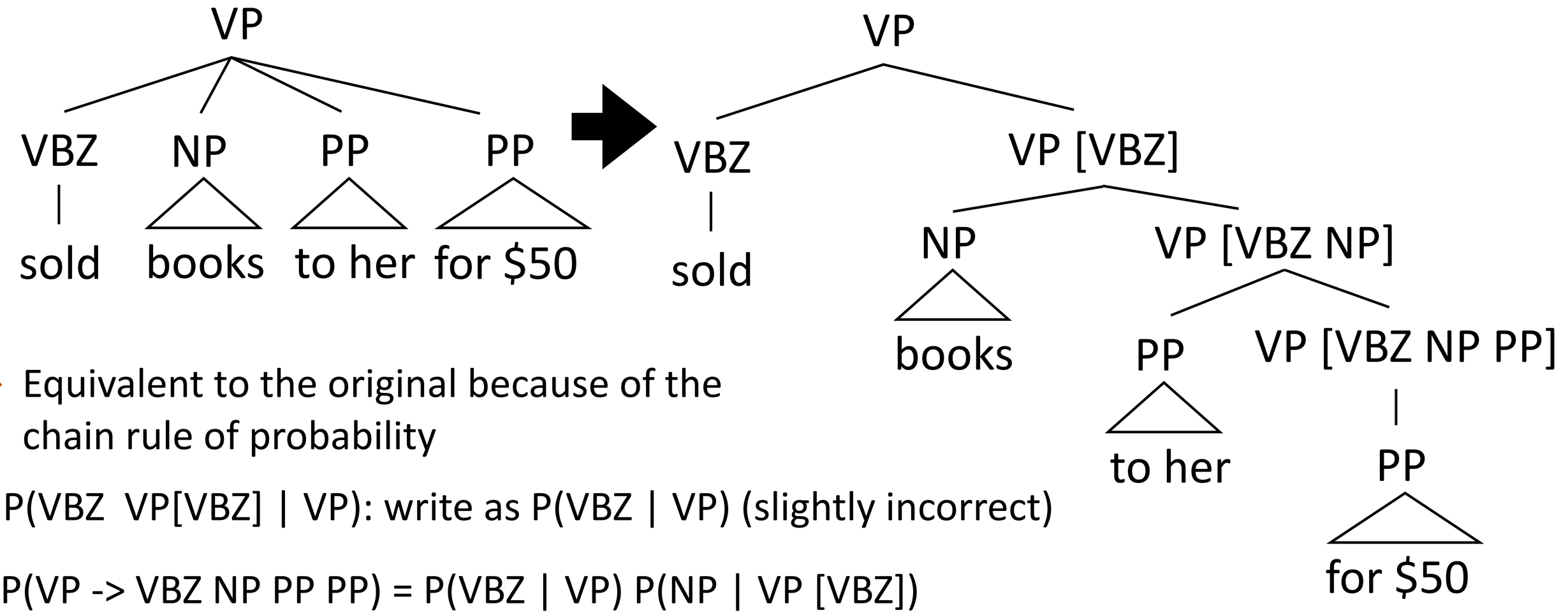


v = 2 Markovization

- Why is this a good idea?

Binarization Revisited

- Another way of doing lossless binarization:

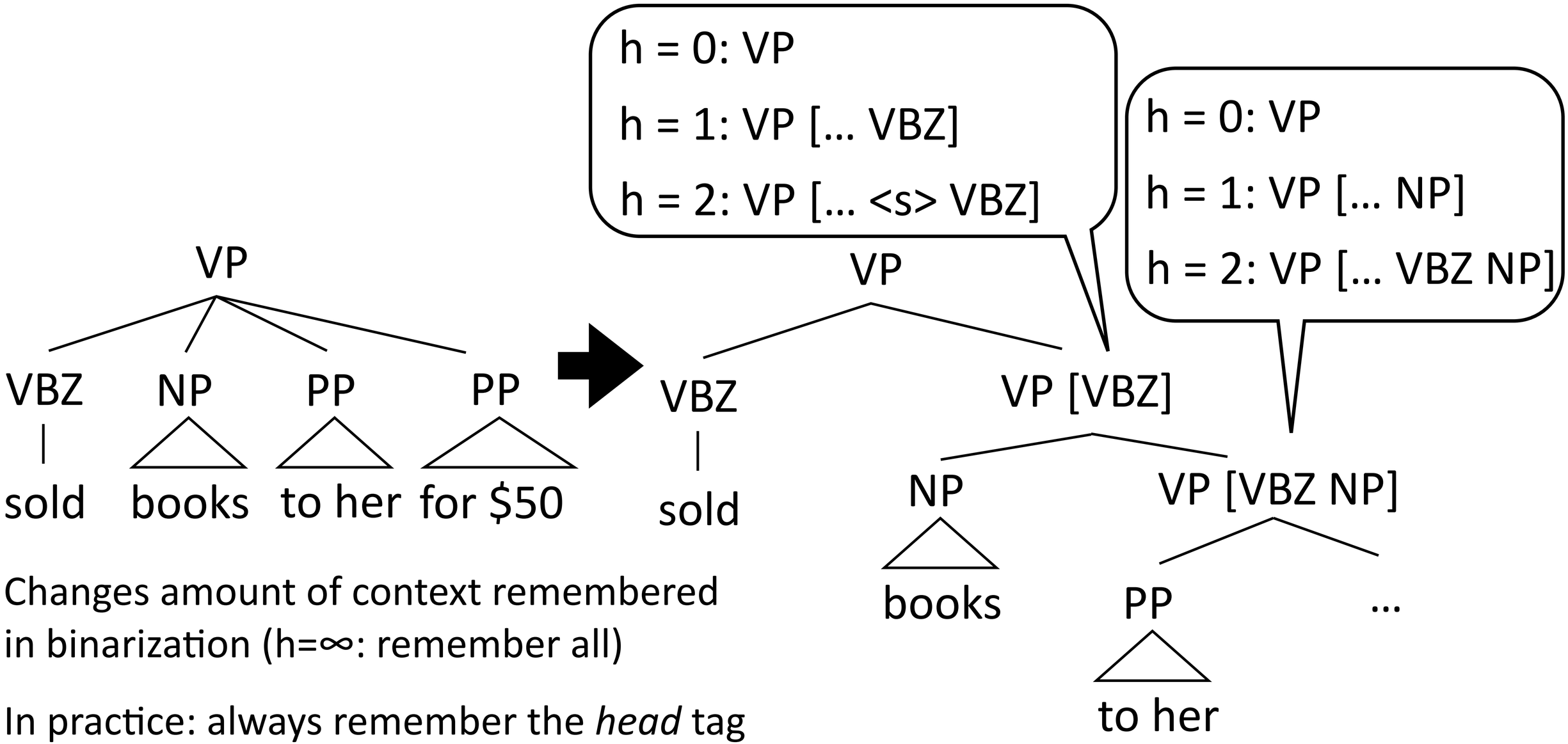


- Equivalent to the original because of the chain rule of probability

P(VBZ VP[VBZ] | VP): write as P(VBZ | VP) (slightly incorrect)

P(VP -> VBZ NP PP PP) = P(VBZ | VP) P(NP | VP [VBZ])
P (PP | VP [VBZ NP]) P (PP | VP [VBZ NP PP])

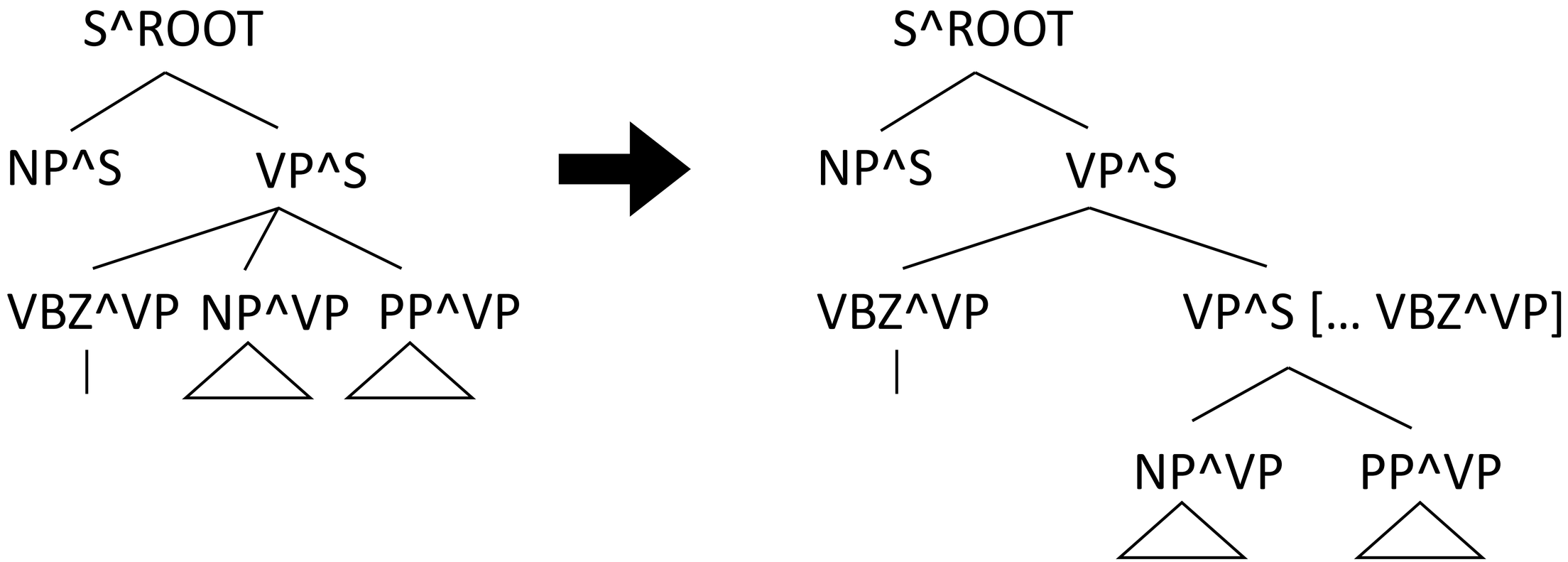
Horizontal Markovization



- Changes amount of context remembered in binarization (h=∞: remember all)
- In practice: always remember the *head* tag

Annotating Trees

- First apply vertical Markovization, then binarize + apply horizontal



Annotating Trees

Vertical Order		Horizontal Markov Order $h = 0$	$h = 1$	$h \leq 2$	$h = 2$	$h = \infty$
$v = 1$	No annotation	71.27 (854)	72.5 (3119)	73.46 (3863)	72.96 (6207)	72.62 (9657)
$v \leq 2$	Sel. Parents	74.75 (2285)	77.42 (6564)	77.77 (7619)	77.50 (11398)	76.91 (14247)
$v = 2$	All Parents	74.68 (2984)	77.42 (7312)	77.81 (8367)	77.50 (12132)	76.81 (14666)
$v \leq 3$	Sel. GParents	76.50 (4943)	78.59 (12374)	79.07 (13627)	78.97 (19545)	78.54 (20123)
$v = 3$	All GParents	76.74 (7797)	79.18 (15740)	79.74 (16994)	79.07 (22886)	78.72 (22002)

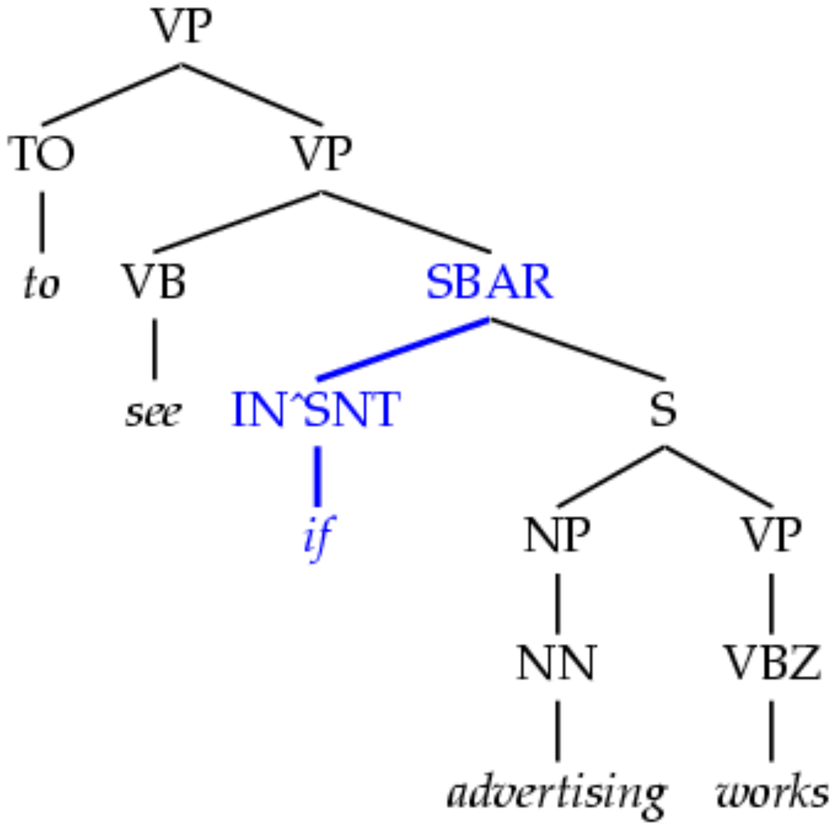
Figure 2: Markovizations: F_1 and grammar size.

Klein and Manning (2003)

Tag Splits

- Can do some other specialized tag splits: e.g., sentential prepositions behave differently from other prepositions

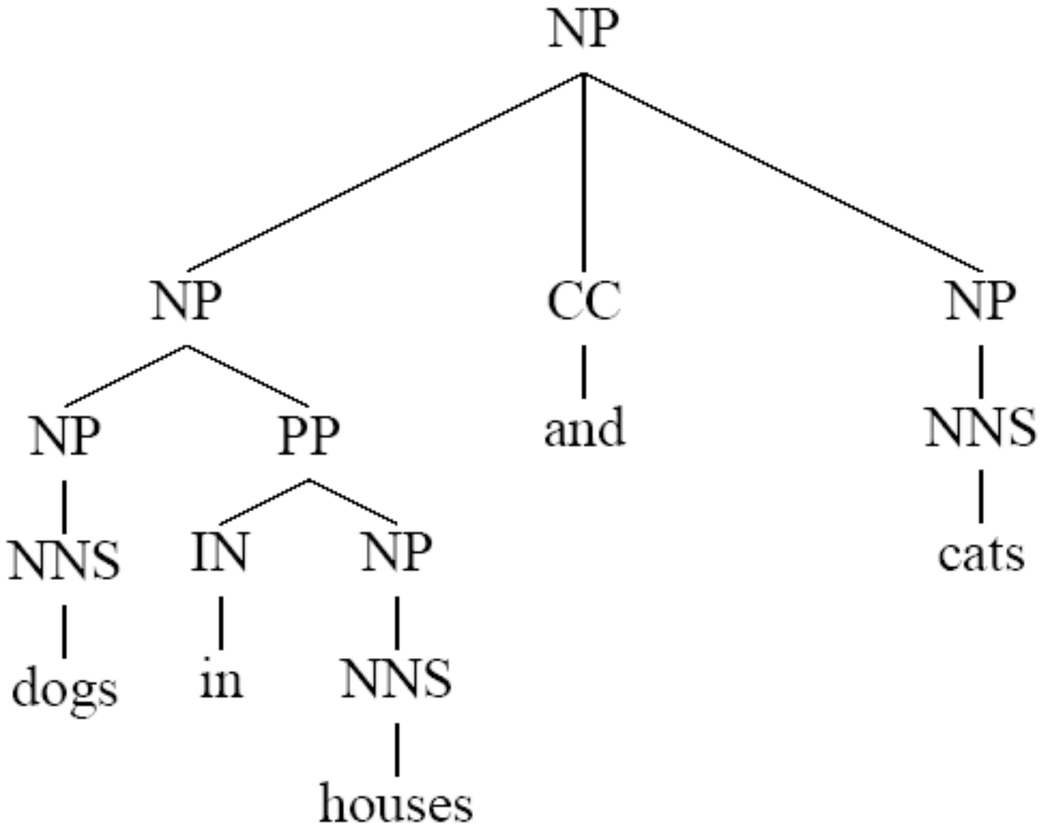
- 79 F1 => 86.3 F1 using more tricks

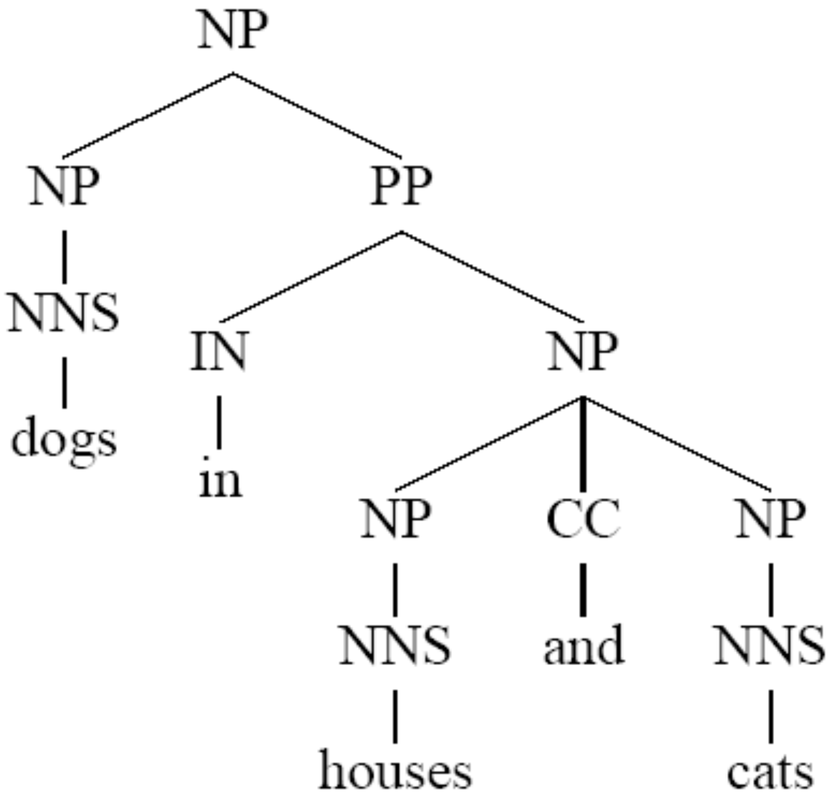


Klein and Manning (2003)

Other Parsers

Lexicalized Parsers





- Even with parent annotation, these trees have the same rules. Need to use the words

Lexicalized Parsers

- Annotate each grammar symbol with its “head word”: most important word of that constituent
- Rules for identifying headwords (e.g., the last word of an NP before a preposition is typically the head)
- Collins and Charniak (late 90s): ~89 F1 with these