PCFG: **P**robabilistic **C**ontext **F**ree **G**rammars

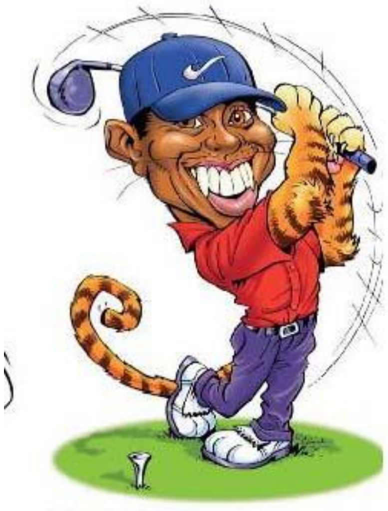
Presenter: Ba Dat Nguyen
Advisor: Dr. Martin Theobald

Max-Planck-Institut für Informatik
Saarbrücken, Germany

Outline

- Introduction
- **P**robabilistic **C**ontext **F**ree **G**rammars
  - Parsing
  - **C**ontext **F**ree **G**rammars
  - **P**robabilistic **C**ontext **F**ree **G**rammars
  - Inside-Outside Algorithm
- Extension
  - Distance
  - Complement/ adjunct distinction
  - Traces and Wh-movement

The World is a big ambiguity



Solution

PCFG is a good way to solve ambiguity problems in syntactic structure field.

Outline

- Introduction
- Probabilistic Context Free Grammars
  - Parsing
  - **C**ontext **F**ree **G**rammars
  - **P**robabilistic **C**ontext **F**ree **G**rammars
  - Inside-Outside Algorithm
- Extension
  - Distance
  - Complement/ adjunct distinction
  - Traces and Wh-movement

max planck institut informatik

Language and Grammar

- Language
  - Structural
  - Ambiguous
- Grammar
  - Generalization of regularities in language structures
  - Morphology and syntax

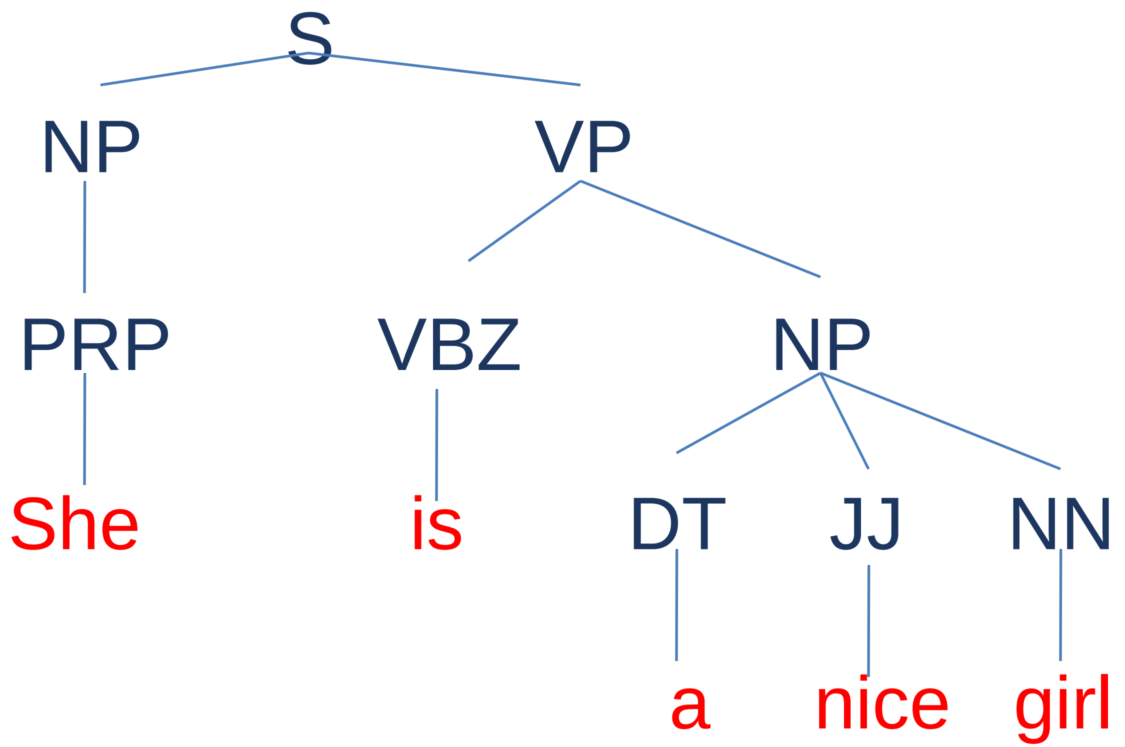
max planck institut informatik

Parsing

- Process working out the grammatical structure of sentences.
- Basic Parsing Algorithms
  - Parsing Strategies
  - CYK Algorithm
  - Earley Algorithm

Example of parsing

- “She is a nice girl”



Outline

- Introduction
- Probabilistic Context Free Grammars
  - Parsing
  - **C**ontext **F**ree **G**rammars
  - **P**robabilistic **C**ontext **F**ree **G**rammars
  - Inside-Outside Algorithm
- Extension
  - Distance
  - Complement/ adjunct distinction
  - Traces and Wh-movement

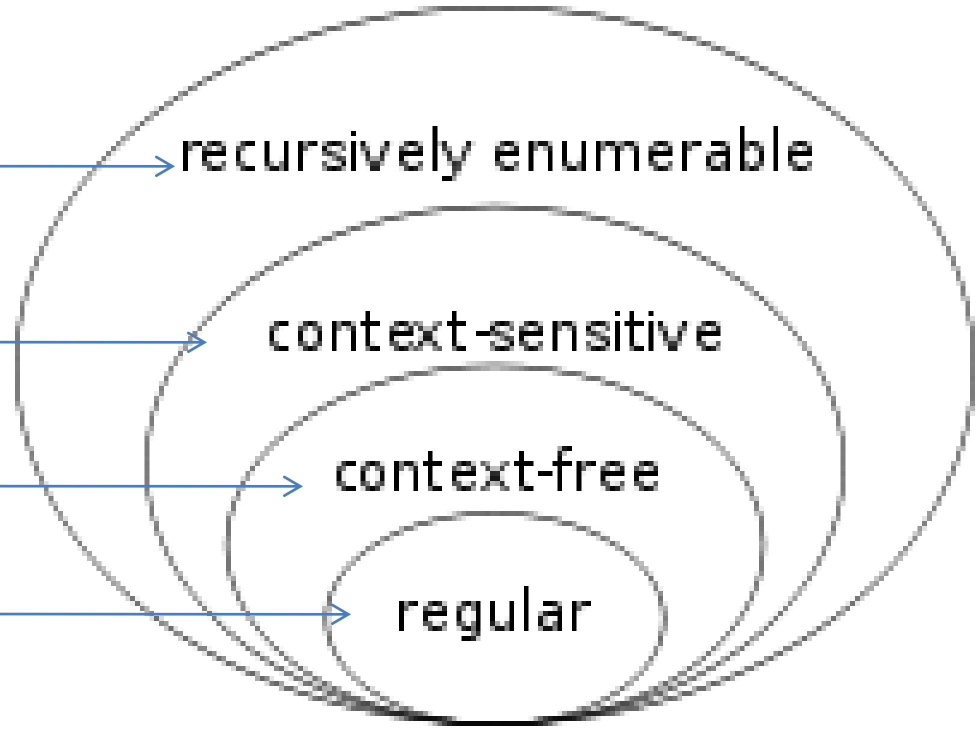
Chomsky hierarchy

$$\alpha \rightarrow \beta$$

$$\alpha A \beta \rightarrow \alpha \gamma \beta$$

$$A \rightarrow \gamma$$

$$A \rightarrow a \ \text{ or } \ A \rightarrow aB$$



Where :

A, B are nonterminals

a is a terminal

α, β, γ are strings of terminals and nonterminals

Context **F**ree **G**rammars (CFG)

- A **C**ontext **F**ree **G**rammars consists of
  - A set of terminals $\{w^k\}$, k = 1,... V
  - A set of nonterminals $\{N^i\}$, i = 1,... n
  - A designated start symbol N^1
  - A set of rules $\{N^i \rightarrow \xi^j\}$
    where ξ^j is a sequence of terminals and nonterminals

Example of CFG

S -> NP VP

PP -> P NP

VP -> V NP

VP -> VP PP

P -> with

V -> saw

NP -> NP PP

NP -> astronomers

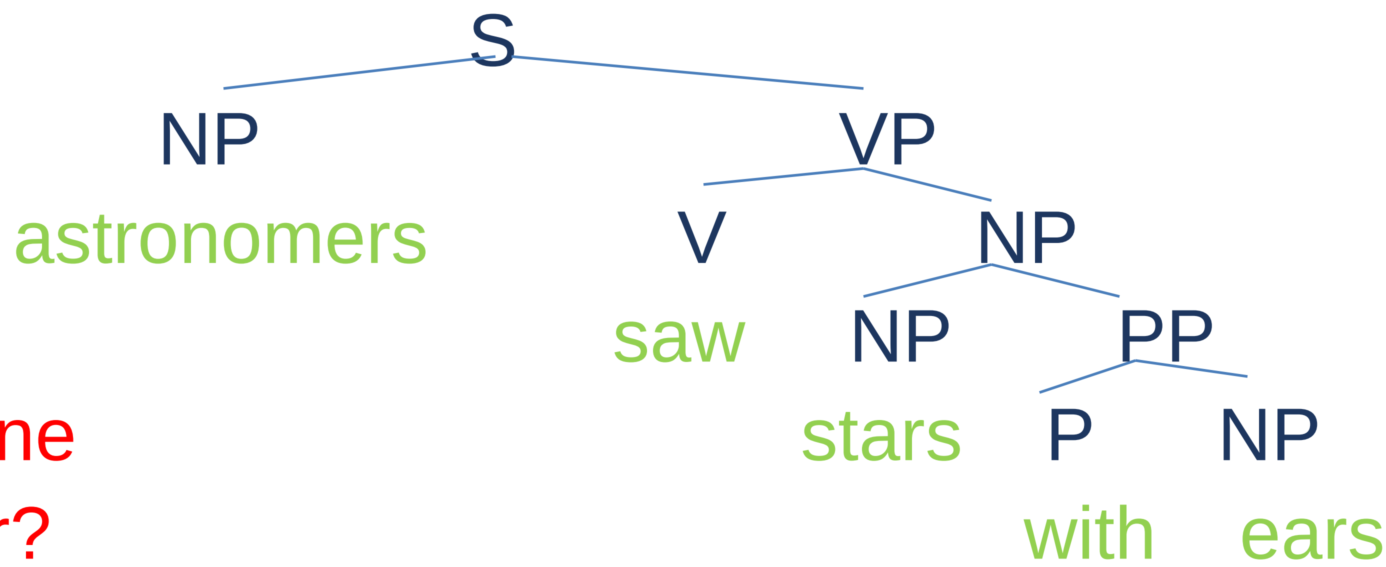
NP -> ears

NP -> saw

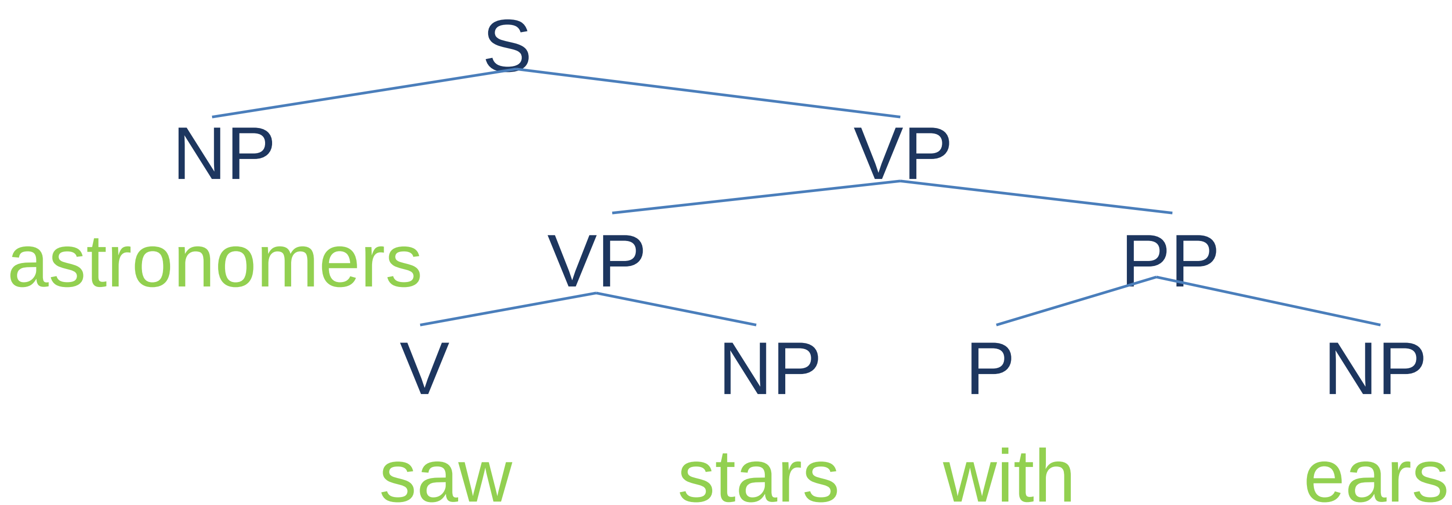
NP -> stars

NP -> telescopes

Ambiguous sentences



Which one
is better?



max planck institut informatik

Outline

- Introduction
- Probabilistic Context Free Grammars
  - Parsing
  - Context Free Grammars
  - **P**robabilistic **C**ontext **F**ree **G**rammars
  - Inside-Outside Algorithm
- Extension
  - Distance
  - Complement/ adjunct distinction
  - Traces and Wh-movement

Probabilistic **CFG**

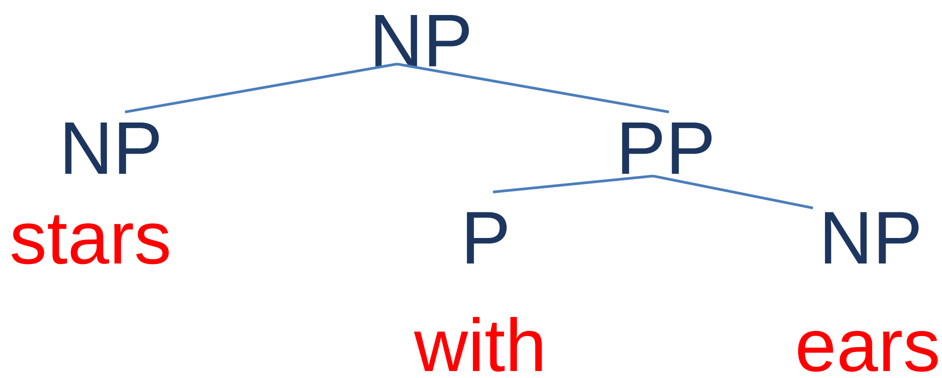
- A **P**robabilistic **C**ontext **F**ree **G**rammars (PCFG) consists of
  - A CFG
  - A corresponding set of probabilities on rules such that:

$$\sum_j P(N^i \rightarrow \xi^j) = 1 \qquad \forall \boldsymbol{i}$$

Example of PCFG

S -> NP VP	1.0	NP -> NP PP	0.4
PP -> P NP	1.0	NP -> astronomers	0.1
VP -> V NP	0.7	NP -> ears	0.18
VP -> VP PP	0.3	NP -> saw	0.04
P -> with	1.0	NP -> stars	0.18
V -> saw	1.0	NP -> telescopes	0.1

Probability of a tree

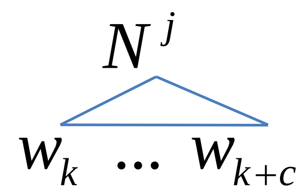


$$
\begin{aligned}
&P(NP \rightarrow NP\ PP, NP \rightarrow stars, PP \rightarrow P\ \mathrm{NP}, P \rightarrow with, NP \rightarrow ears) \\
&= P(S \rightarrow NP\ PP) \times P(NP \rightarrow stars \mid NP \rightarrow NP\ PP) \\
&\times P(PP \rightarrow P\ \mathrm{NP} \mid NP \rightarrow NP\ PP, NP \rightarrow stars) \\
&\times P(P \rightarrow with \mid PP \rightarrow P\ \mathrm{NP}, NP \rightarrow NP\ PP, NP \rightarrow stars) \\
&\times P(NP \rightarrow ears \mid P \rightarrow with, PP \rightarrow P\ \mathrm{NP}, NP \rightarrow NP\ PP, NP \rightarrow stars)
\end{aligned}
$$

Assumptions

- Place invariance

$$\forall k \;\; P(N^j_{k(k+c)} \rightarrow \xi) \;\; \textit{is the same}$$



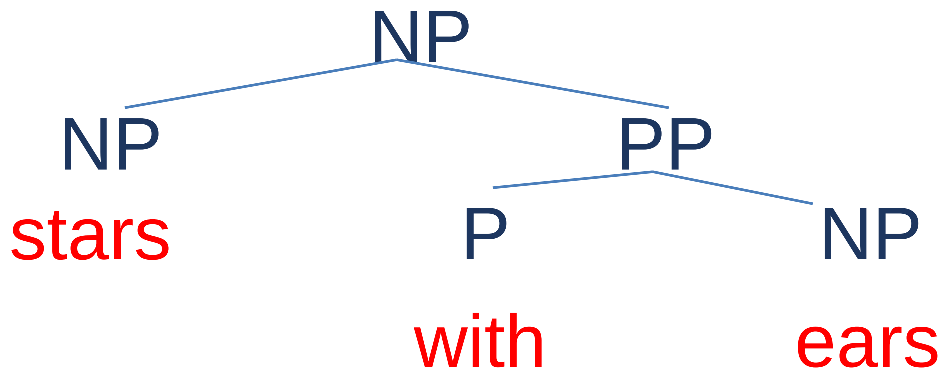
- Context-free

$$P(N^j_{kl} \rightarrow \xi \,|\, \textit{anything outside k through l}) \;=\; P(N^j_{kl} \rightarrow \xi)$$

- Ancestor-free

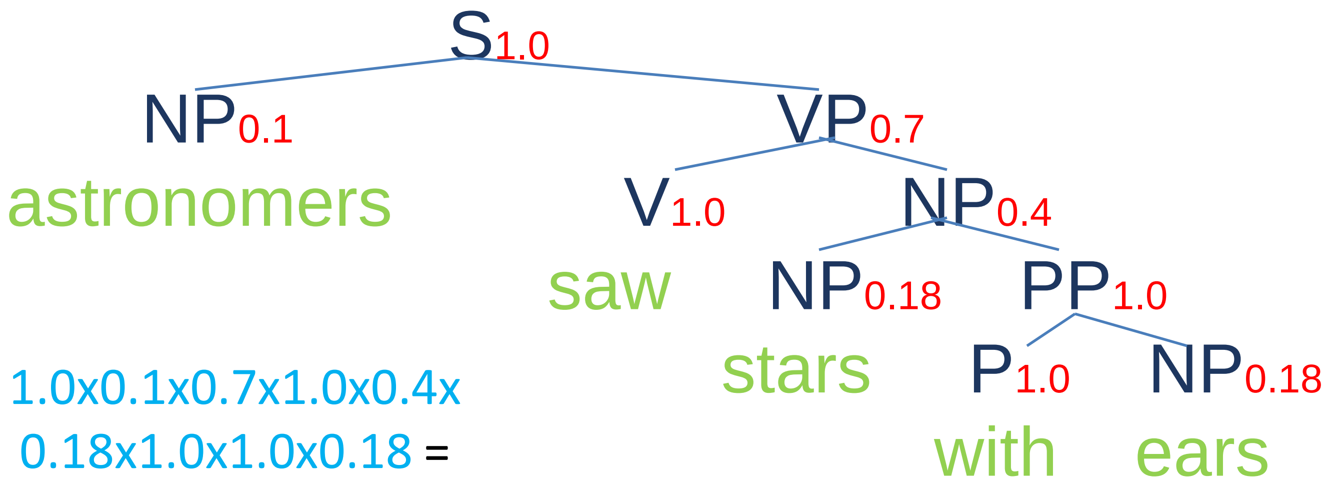
$$P(N^j_{kl} \rightarrow \xi \,|\, \textit{any ancestor nodes outside } N^j_{kl}) \;=\; P(N^j_{kl} \rightarrow \xi)$$

Probability of a tree

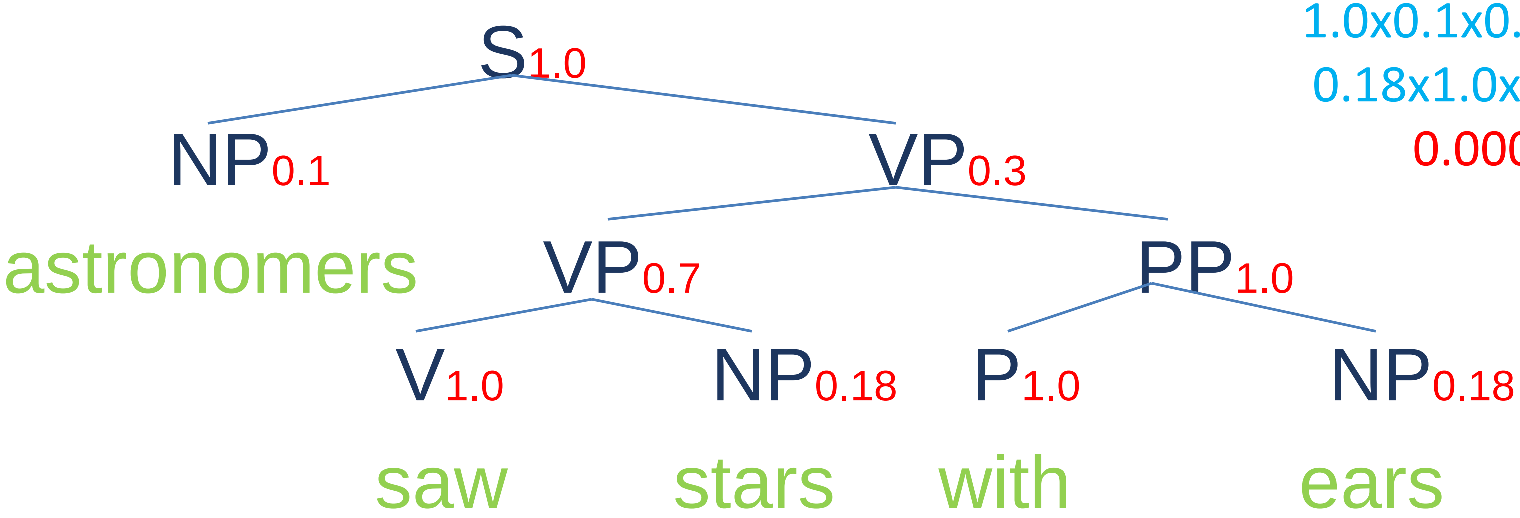


$$P(NP \rightarrow NP\ PP,\ NP \rightarrow stars,\ PP \rightarrow P\ \mathrm{NP},\ P \rightarrow with,\ NP \rightarrow ears)$$
$$= P(S \rightarrow NP\ PP) \times P(NP \rightarrow stars) \times P(PP \rightarrow P\ \mathrm{NP})$$
$$\times P(P \rightarrow with) \times P(NP \rightarrow ears)$$

Ambiguity



1.0x0.1x0.7x1.0x0.4x
0.18x1.0x1.0x0.18 =
0.0009072



1.0x0.1x0.3x0.7x1.0x
0.18x1.0x1.0x0.18 =
0.0006804

Outline

- Introduction
- Probabilistic Context Free Grammars
  - Parsing
  - **C**ontext **F**ree **G**rammars
  - **P**robabilistic **C**ontext **F**ree **G**rammars
  - Inside-Outside Algorithm
- Extension
  - Distance
  - Complement/ adjunct distinction
  - Traces and Wh-movement

Probability of a rule

- Given a training set of annotated sentences

$$P(N^j \rightarrow \xi) = \frac{C(N^j \rightarrow \xi)}{\sum_{\gamma} C(N^j \rightarrow \gamma)}$$

C(.) - number of times that a particular rule is used.

Probability of a rule

How to calculate if there is no annotated data!

Maximum Likelihood Estimation

- Maximum Likelihood Estimation

$$\arg\max_{\mu} P(O_{training} \mid \mu)$$

$$\mu = \text{parameters of current grammar set}$$

- No known analytic method to choose μ to maximize P(O | μ)
- Locally maximize P(O | μ) by an iterative hill-climbing – special case of **E**xpectation **M**aximization method.
- Inside-Outside algorithm is a form of EM using the inside-outside probabilities estimated from training set.

Training a PCFG

- We are given
  - A set of training sentences
  - A set of terminals
  - A set of nonterminals
- Initial probabilities are estimated by rules (perhaps by randonly chosen)
- Using inside-outside algorithm to train

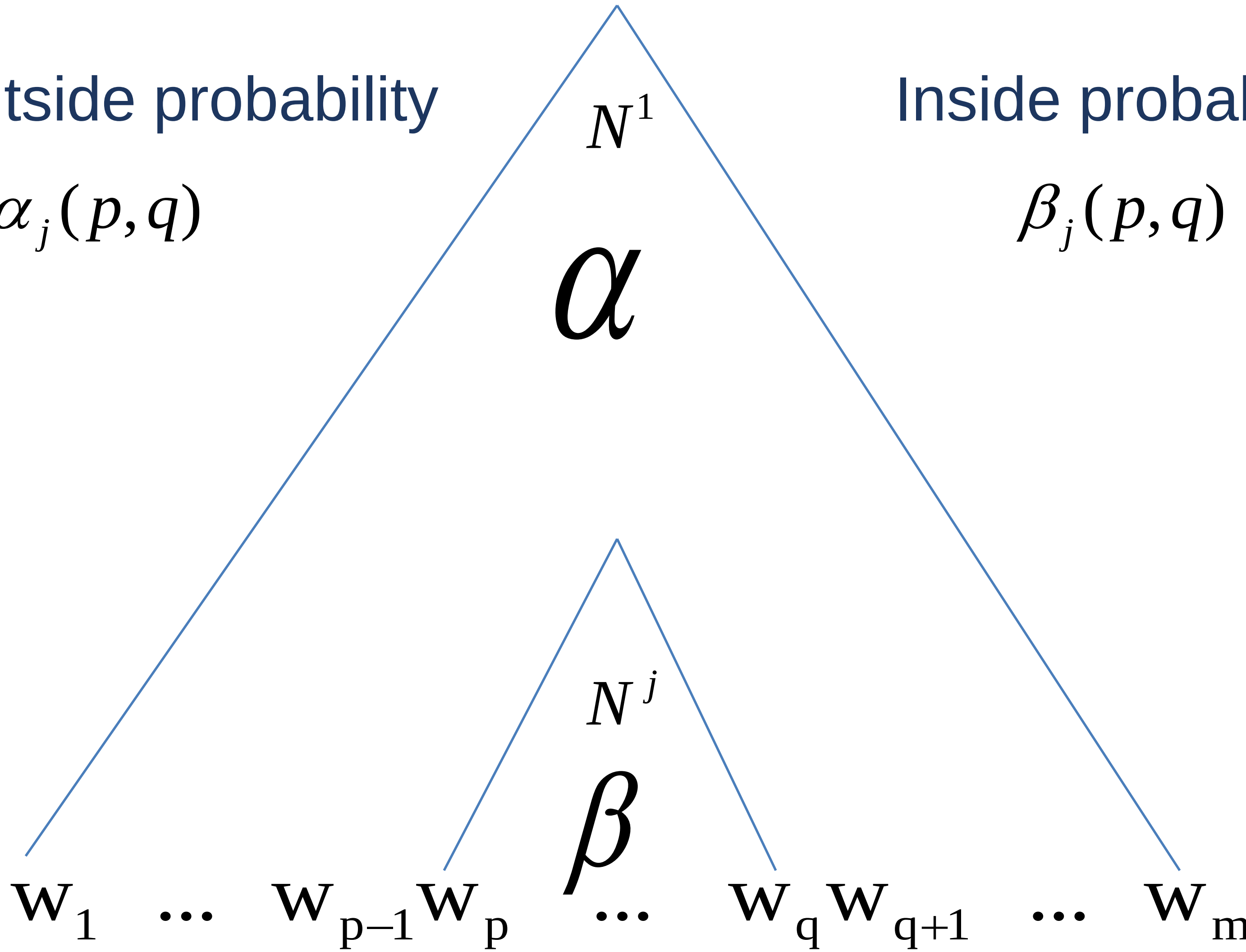
Inside-Outside probabilities

- Outside probability

  $\alpha_j(p,q)$

Inside probability

$\beta_j(p,q)$



Inside probabilities

- Inside probability $\beta_j(p,q)$ is the probability of sequence $w_p \dots w_q$ being generated with a tree rooted by node N^j

$$\beta_j(p,q) = P(w_{pq} \mid N^j_{pq})$$

- Calculation can be carried out bottom-up

$$\beta_j(k,k) = P(N^j \to w_k)$$

$$\beta_j(p,q) = \sum_{r,s} \sum_{d=q}^{q-1} P(N^j \to N^r N^s) \beta_r(p,d) \beta_s(d+1,q)$$

Outside probabilities

- Outside probability $\alpha_j(p,q)$ is the total probability of beginning with the start symbol and generating all the words outside N^j_{pq}

$$\alpha_j(p,q) = P(w_{1,p-1}, N^j_{pq}, w_{q+1,n})$$

$$N^j_{pq} = N^j \xrightarrow{*} w_p \dots w_q$$

$$\alpha_1(1,m) = 1 \; and \; \alpha_j(1,m) = 0 \; for \; j \neq 1$$

$$\alpha_j(p,q) = \sum_{f,g} \sum_{e=q+1}^{m} \alpha_f(p,e) P(N^f \rightarrow N^j N^g) \beta_g(q+1,e)$$

$$+ \sum_{f,g} \sum_{e=1}^{p-1} \alpha_f(e,q) P(N^f \rightarrow N^g N^j) \beta_g(e,p-1)$$

Inside-Outside Algorithm

We have:

$$\alpha_j(p,q)\beta_j(p,q) = P(N^1 \xrightarrow{*} w_{1,m}, N^j \xrightarrow{*} w_{p,q})$$

$$Call\ P(N^1 \xrightarrow{*} w_{1,m})\ \pi$$

$$P(N^j \xrightarrow{*} w_{p,q} \mid N^1 \xrightarrow{*} w_{1,m}) = \frac{\alpha_j(p,q)\beta_j(p,q)}{\pi}$$

Inside-Outside Algorithm

$$E(N^j \text{ is used}) = \frac{\sum_{p=1}^{m}\sum_{q=p}^{m}\alpha_j(p,q)\beta_j(p,q)}{\pi}$$

$$E(N^j \to N^r N^s, N^j \text{ used})$$

$$= \frac{\sum_{p=1}^{m-1}\sum_{q=p+1}^{m}\sum_{d=p}^{q-1}\alpha_j(p,q)P(N^j \to N^r N^s)\beta_r(p,d)\beta_s(d+1,q)}{\pi}$$

Inside-Outside Algorithm

$$Therefore:$$

$$P(N^j \rightarrow N^r N^s) = \frac{\sum_{p=1}^{m-1} \sum_{q=p+1}^{m} \sum_{d=p}^{q-1} \alpha_j(p,q) P(N^j \rightarrow N^r N^s) \beta_r(p,d) \beta_s(d+1,q)}{\sum_{p=1}^{m} \sum_{q=1}^{m} \alpha_j(p,q) \beta_j(p,q)}$$

$$P(N^j \rightarrow w^k) = \frac{\sum_{h=1}^{m} \alpha_j(h,h) P(w_h = w^k) \beta_j(h,h)}{\sum_{p=1}^{m} \sum_{q=1}^{m} \alpha_j(p,q) \beta_j(p,q)}$$

Inside-Outside Algorithm

For each sentence W^i in the corpus

$$f_i(p,q,j,r,s) = \frac{\sum_{d=p}^{q-1} \alpha_j(p,q) P(N^j \rightarrow N^r N^s) \beta_r(p,d) \beta_s(d+1,q)}{P(N^1 \xrightarrow{*} W_i)}$$

$$g_i(h,j,k) = \frac{\alpha_j(h,h) P(w_h = w^k) \beta_j(h,h)}{P(N^1 \xrightarrow{*} W_i)}$$

$$h_i(p,q,j) = \frac{\alpha_j(p,q) \beta_j(p,q)}{P(N^1 \xrightarrow{*} W_i)}$$

Inside-Outside Algorithm

We have

$$P(N^j \rightarrow N^r N^s) = \frac{\sum_{i=1}^{l} \sum_{p=1}^{m_i-1} \sum_{q=p+1}^{m_i} f_i(p,q,j,r,s)}{\sum_{i=1}^{l} \sum_{p=1}^{m_i} \sum_{q=p}^{m_i} h_i(p,q,j)}$$

$$P(N^j \rightarrow w^k) = \frac{\sum_{i=1}^{l} \sum_{h=1}^{m_i} g_i(h,j,k)}{\sum_{i=1}^{l} \sum_{p=1}^{m_i} \sum_{q=p}^{m_i} h_i(p,q,j)}$$

Discussion

- Inside-Outside algorithm is quite slow $O(m^3n^3)$ for each sentence
  - m is the length of the sentence
  - n is the number of nonterminals
- The algorithm is very sensitive to the initialization of the parameters.
- In practice, a PCFG is a worse language model for English than an n-gram model (n>1).

Outline

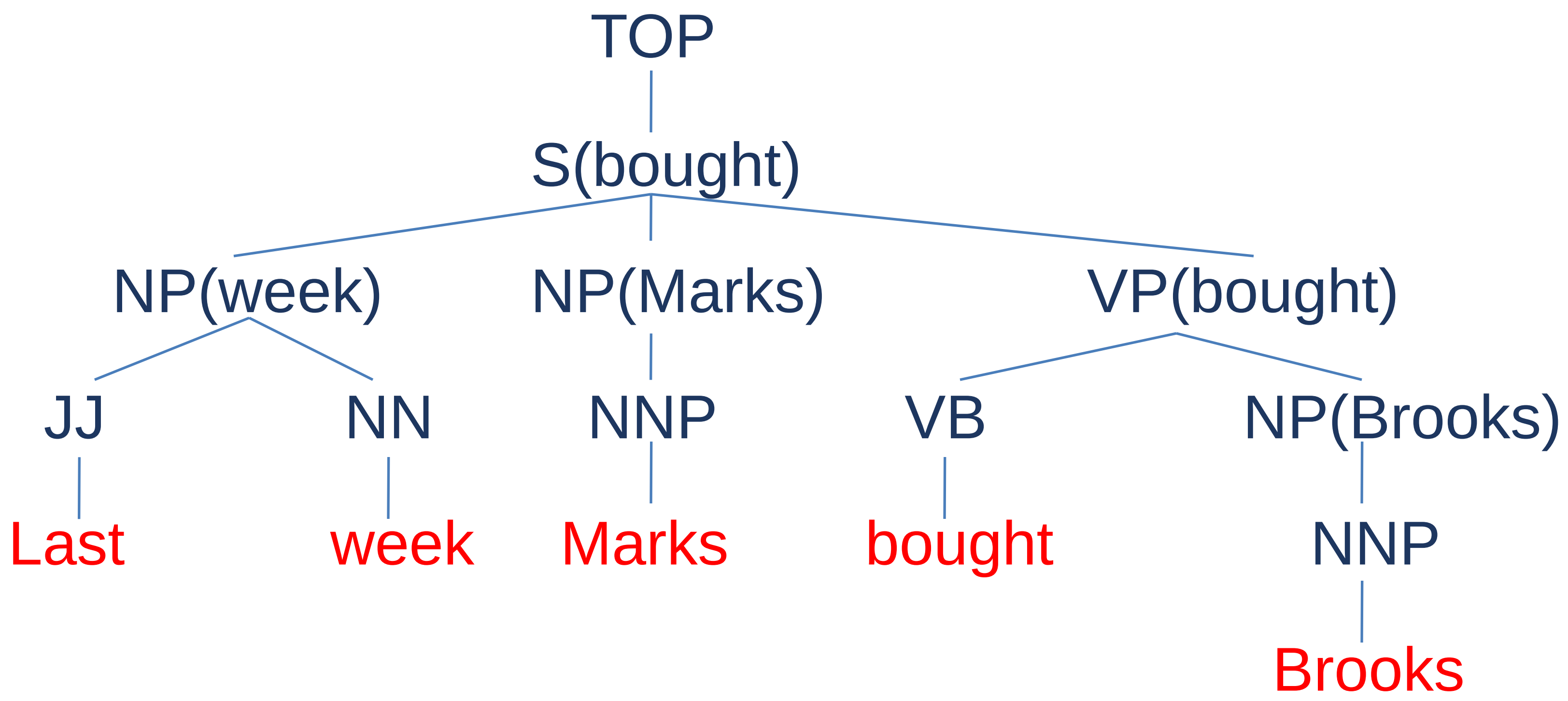
- Introduction
- Probabilistic Context Free Grammars
  - Parsing
  - Context Free Grammars
  - Probabilistic Context Free Grammars
  - Inside-Outside Algorithm
- Extension
  - Distance
  - Complement/ adjunct distinction
  - Traces and Wh-movement

More features

TOP	->	S(bought)		
S(bought)	->	NP(week)	NP(Marks)	VP(bought)
NP(week)	->	JJ(Last)	NN(week)	
NP(Marks)	->	NNP(Marks)		
VP(bought)	->	VB(bought)	NP(Brooks)	
NP(Brooks)	->	NNP(Brooks)		

- Adding some lexical features: words and POS inside non-terminals.
- Using the head-child of the phrase, which inherits the head-word from its parent.

Example


TOP
S(bought)
NP(week)
NP(Marks)
VP(bought)
JJ
NN
NNP
VB
NP(Brooks)
Last
week
Marks
bought
NNP
Brooks

Using Distance

- Using some probabilities
  - Head constituent label of the phrase

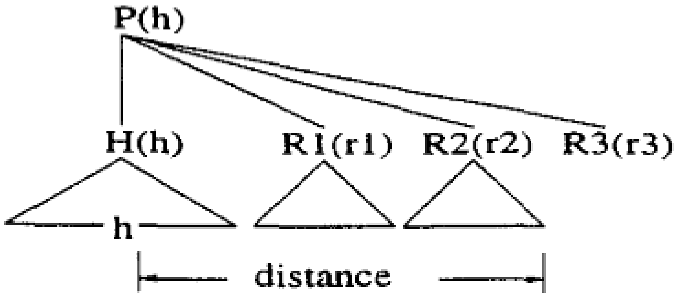
$$P_H(H \mid P, h)$$

  - Modifiers to the right of the head

$$\prod_{i=1..m+1} P_R(R_i(r_i) \mid P, h, H, R_1(r_1)...R_{i-1}(r_{i-1}))$$

  - Modifiers to the left of the head

$$\prod_{i=1..n+1} P_L(L_i(l_i) \mid P, h, H, L_1(l_1)...L_{i-1}(l_{i-1}))$$



Using Distance

$$For\ example:$$

$$P\,(S(bought) \rightarrow NP(week)NP(Marks)VP(bought))$$

$$= P_h(VP \mid S, bought) \times P_l(NP(Marks) \mid S, VP, bought)$$

$$\times P_l(NP(week) \mid S, VP, bought, NP, Marks)$$

$$\text{With distance } 0$$

$$P\,(S(bought) \rightarrow NP(week)NP(Marks)VP(bought))$$

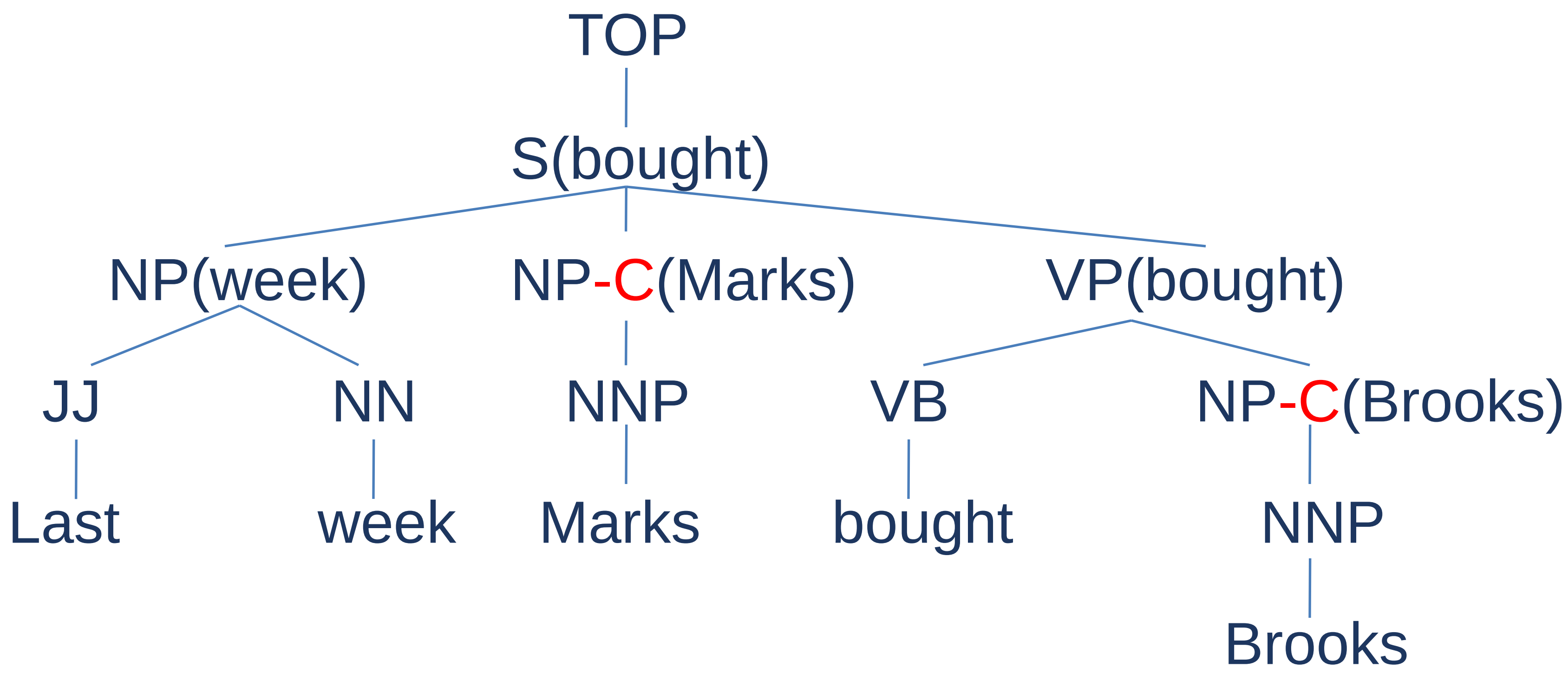
$$= P_h(VP \mid S, bought) \times P_l(NP(Marks) \mid S, VP, bought)$$

$$\times P_l(NP(week) \mid S, VP, bought)$$

Outline

- Introduction
- Probabilistic Context Free Grammars
  - Parsing
  - Context Free Grammars
  - Probabilistic Context Free Grammars
  - Inside-Outside Algorithm
- Extension
  - Distance
  - Complement/ adjunct distinction
  - Traces and Wh-movement

Adding the complement / adjunct distinction



It would be useful to identify "Marks" as a subject and "Last week" as an adjunct!

Adding the complement / adjunct distinction

- Adding “-C“ suffix to all non-terminals in training data which satisfy:
  - The nonterminal must be: an NP, SBAR or S whose parent is an S; an NP, SBAR, S, or VP whose parent is a VP; or S whose parent is an SBAR
  - The non-terminal must not have one of the following semantic tags: ADV, VOC, BNF, DIR, EXT, LOC, MNR, TMP, CLR or PRP.

Adding the complement / adjunct distinction

- Using some probabilities
  - Head constituent label of the phrase $P_H(H \mid P, h)$
  - Left and right subcat frames $P_{lc}(LC \mid P, H, h)$ *and* $P_{rc}(RC \mid P, H, h)$
  - Modifiers to the right of the head $P_R(R_i, r_i \mid H, P, h, \text{distance}(i-1), RC)$
  - Modifiers to the left of the head $P_L(L_i, l_i \mid H, P, h, \text{distance}(i-1), LC)$

Adding the complement / adjunct distinction

- For example

$$P(S(bought) \to NP(week)\ NP-C(Marks)\ VP(bought))$$
$$= P_h(VP \mid S, bought)$$
$$\times P_{lc}(\{NP-C\} \mid S, VP, bought)$$
$$\times P_l(\{NP-C(Marks)\} \mid S, VP, bought, \{NP-C\})$$
$$\times P_l(NP(week) \mid S, VP, bought)$$

Outline

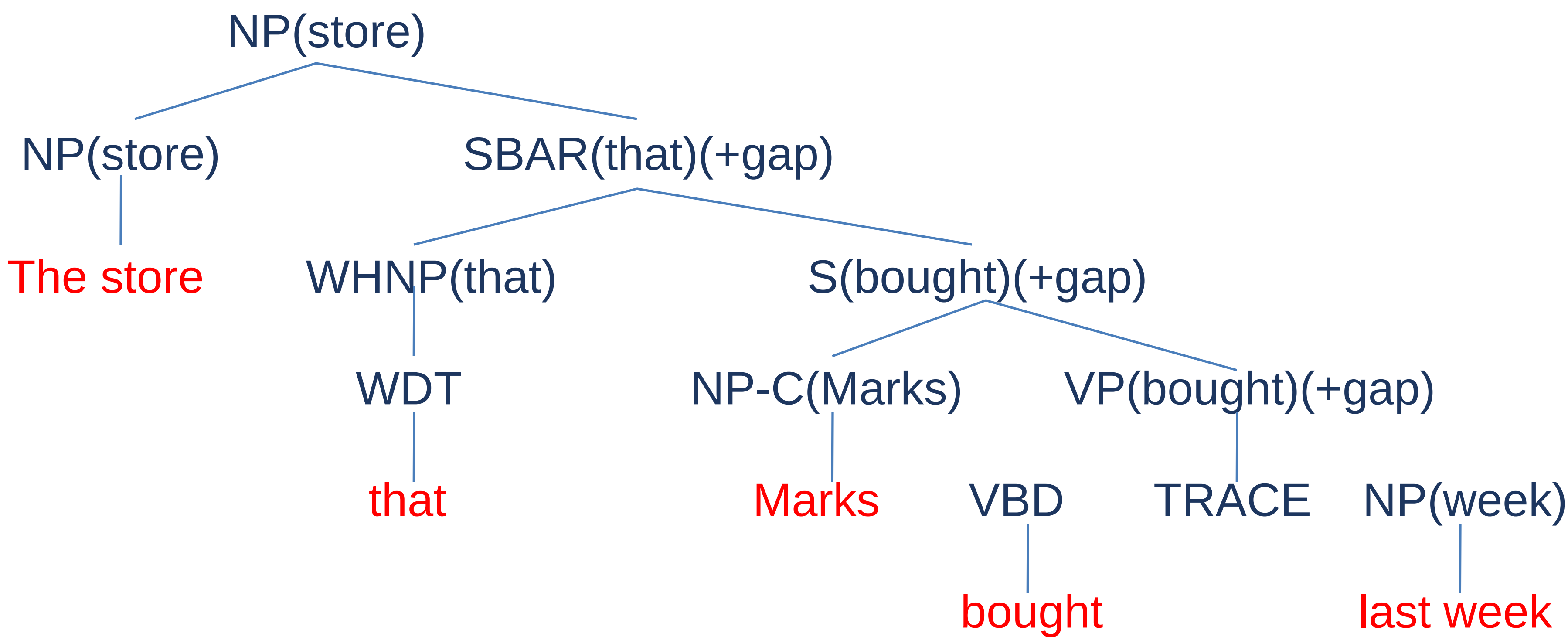
- Introduction
- Probabilistic Context Free Grammars
  - Parsing
  - Context Free Grammars
  - Probabilistic Context Free Grammars
  - Inside-Outside Algorithm
- Extension
  - Distance
  - Complement/ adjunct distinction
  - Traces and Wh-movement

Traces and Wh-movement

- Adding a “*gap*“ feature to each non-terminal in the tree and propagating gaps through the tree until the are finally discharged as a trace complement.
- For example

(1)	NP	->	NP	SBAR(+gap)
(2)	SBAR(+gap)	->	WHNP	S-C(+gap)
(3)	S(+gap)	->	NP-C	VP(+gap)
(4)	VP(+gap)	->	VB	TRACE NP

Traces and Wh-movement



$$P_l(VP(bought)(+gap) \rightarrow VB(bought)\ TRACE\ NP(week))$$

$$= P_h(VB \mid VP, bought) \times P_G(Right \mid VP, bought, VB)$$

$$\times P_{RC}(\{NP - C\} \mid VP, bought, VB)$$

$$\times P_R(TRACE \mid VP, bought, VB, \{NP - C, +gap\})$$

$$\times P_R(NP(week) \mid VP, bought, VB)$$

max planck institut informatik

Experiment

Models are trained on sections 02 - 21 of the Wall Street Journal portion of the Penn Treebank and tested on section 23.

Labelled Precision =

$$\frac{\text{number of correct constituents in proposed parse}}{\text{number of constituents in proposed parse}}$$

Labelled recall =

$$\frac{\text{number of correct constituents in proposed parse}}{\text{number of constituents in treebank parse}}$$

Crossing Brackets = number of constituents which violate constituent boundaries with a constituent in the treebank parse

MODEL	≤ 40 Words (2245 sentences)					≤ 100 Words (2416 sentences)				
	LR	LP	CBs	0 CBs	≤ 2 CBs	LR	LP	CBs	0 CBs	≤ 2 CBs
(Magerman 95)	84.6%	84.9%	1.26	56.6%	81.4%	84.0%	84.3%	1.46	54.0%	78.8%
(Collins 96)	85.8%	86.3%	1.14	59.9%	83.6%	85.3%	85.7%	1.32	57.2%	80.8%
Model 1	87.4%	88.1%	0.96	65.7%	86.3%	86.8%	87.6%	1.11	63.1%	84.1%
Model 2	88.1%	88.6%	0.91	66.5%	86.9%	87.5%	88.1%	1.07	63.9%	84.6%
Model 3	88.1%	88.6%	0.91	66.4%	86.9%	87.5%	88.1%	1.07	63.9%	84.6%

References

1. Foundations of Statistical Natural Language Processing
2. Standford parse http://nlp.stanford.edu:8080/parser/
3. Three Generative, Lexicalised Models for Statistical Parsing, ACL 97

Thanks!