Syntactic analysis

Complexities and search strategies

Parsing algorithms

PROBLEMS WITH NATURAL LANGUAGE ANALYSIS (here: syntactic parsing)

Ambiguity – is typically heavier than intuitively expected

Structural (e.g., attachment of prepositional phrases, coordination)

Lexical (e.g., word forms that can be interpreted as different categories)

Morphological (e.g., underspecified in terms of casus)

Coverage

Large-scale grammars with sufficient coverage of a natural language

Efficiency

How to guide the analysis of a natural language sentence effectively

SOME EXAMPLES OF AMBIGUOUS ANALYSES

Hinter dem Betrug werden die gleichen Täter vermutet, die während der vergangenen Tage in Griechenland gefälschte Banknoten in Umlauf brachten.
(Behind the fraud, the same actors are assumed, who distributed faked bank notes during the past days in Greece)

Es klappte gut weil Maria die Freundin von Anna aus Osnabrück mit dem Auto von Petra aus Bielefeld abgeholt hat.
(*It worked well because Mary picked up the friend of Ann from Osnabrück with the car of Petra from Bielefeld)*

Es klappte gut weil Maria die Freundin von Anna aus Osnabrück mit dem Auto von Petra aus Bielefeld abgeholt wurde.

(*It worked well because Mary the friend of Ann from Osnabrück was picked up with/by the car of Petra from Bielefeld)*

SOME EXAMPLES OF AMBIGUOUS ANALYSES

Hinter dem Betrug werden die gleichen Täter vermutet, die während der vergangenen Tage in Griechenland gefälschte Banknoten in Umlauf brachten.

92 readings PARGRAM (Kuhn and Rohrer, 1997)

Es klappte gut weil Maria die Freundin von Anna aus Osnabrück mit dem Auto von Petra aus Bielefeld abgeholt hat.

1732 readings GEPARD (Langer, 2001)

Es klappte gut weil Maria die Freundin von Anna aus Osnabrück mit dem Auto von Petra aus Bielefeld abgeholt wurde.

192 readings GEPARD (Langer, 2001)

SEARCH STRATEGIES

Top-down – control by the grammar

Beginning *expansion* with the *start symbol*,

until reaching terminal symbols

Bottom-up – control by the data

Rules applied as reductions, rules interpreted from right to left,

until a reduction to the start symbol is obtained

Left-corner

Combination of bottom-up and top-down

Starting with first element of the right side of a grammar rule

Bi-directional continuation

PROBLEMS WITH CHRONOLOGOUS SEARCHES (I)

Top-down

Builds trees that are not consistent with the input

Left recursion may lead to endless loops

Do badly if there are many different rules for the same left-hand side

Hopeless for rewriting parts of speech (preterminal) with words (terminals)

Always done bottom-up as lexical look-up

Repeated work anywhere there is common substructure

Can work exponential in sentence length

PROBLEMS WITH CHRONOLOGOUS SEARCHES (II)

Bottom-up

Builds trees that may never have a chance to be combined with neighbours

Repeated work anywhere there is common substructure

Inefficient when there is great lexical ambiguity

Unable to deal with empty categories

termination problem unless restricted (but then it is generally incomplete)

Can work exponential in sentence length

USING CHRONOLOGOUS SEARCHES ADEQUATELY

Requirements for combined uses

Exploiting constraints imposed by both grammar and input words

Storing partial results for reuse and recombinations

Principles for success

Left recursion structures must be found, not predicted

Empty categories must be predicted, not found

Alternative ways of fixing problems

Grammar transformation

yields different derivation trees, typically not favored by linguists

LEFT-CORNER – ADDING BOTTOM-UP FILTERING

Idea

Expand grammar off-line to obtain left-corners of rules

Expand only rules that include a left corner consistent with the input

Example

Grammar rules	*Left corners per category*
S -> NP VP | Aux NP VP | VP	**Det, Proper-Noun, Aux, Verb**
NP -> Det NOMINAL | Proper-Noun	**Det, Proper-Noun**
NOMINAL -> Noun | Noun NOMINAL	**Noun**
VP -> Verb | Verb NP	**Verb**

For “Does this flight include a meal?” only the second S rule applies

PARSING ALGORITHMS

Commonalities

Dynamic programming techniques

Systematically filling in tables with solutions to subproblems

Partial solutions are subtrees obtained for sequences of input words

Multiple trees stored to represent ambiguities

Algorithms

Earley [Earley, 1970] – parallel top-down search – O(N^3) for N words

CYK (Cocke, Younger, Kasami) [Kasami 1965] [Younger 1967] – also O(N^3)

CYK ALGORITHM (Recognizer version)

Input

string of n words

Ouput

yes/no (parse tree for the full version)

Data structure

n x n table

rows labeled 0 … n-1, columns 1 … n

cell(i,j) lists possible constituent words spanning between i and j

CYK ALGORITHM (Recognizer version) II

Pseudo code

for ***i*** **:= 1 to** ***n***

    add to (*i*-1,*i*) all (part-of-speech) categories for the *i*th word

for ***width*** **:= 2 to** ***n***

    for ***start*** **:= 0 to** ***n*** **-** ***width***

        end **:=** ***start*** **+** ***width***

        for ***mid*** **:=** ***start*** **+ 1 to** ***end*** **- 1**

            for every constituent *X* in (*start*, *mid*)

                for every constituent *Y* in (*mid*, *end*)

                    for all ways of combining *X* and *Y* (if any)

                        add the resulting constituent to (*start*, *end*)

EXAMPLE

Sentence

“time flies like an arrow” (multiple ambiguities)

Grammar

Terminals	*Non-terminals*
NP -> time	**S -> NP VP**
Vst -> time	**S -> Vst NP**
NP -> flies	**S -> S PP**
VP -> flies	**VP -> V NP**
P -> like	**VP -> VP PP**
V -> like	**NP -> Det N**
Det -> an	**NP -> NP PP**
N -> arrow	**NP -> NP NP**
	PP -> P NP

ANALYSIS (1)

	time 1	flies 2	like 3	an 4	arrow 5
0	NP Vst				
1		NP VP			
2			P V		
3				Det	
4					N

NP -> time
Vst -> time
NP -> flies
VP -> flies
P -> like
V -> like
Det -> N
N -> arrow

S -> NP VP
S -> Vst NP
S -> S PP

VP -> V NP
VP -> VP PP

NP -> Det N
NP -> NP PP
NP -> NP NP

PP -> P NP

ANALYSIS (2)

	time 1	flies 2	like 3	an 4	arrow 5
0	NP Vst				
1		NP VP			
2			P V		
3				Det	
4					N

S -> NP VP
S -> Vst NP
S -> S PP

VP -> V NP
VP -> VP PP

NP -> Det N
NP -> NP PP
NP -> NP NP

PP -> P NP

ANALYSIS (3)

	time 1	flies 2	like 3	an 4	arrow 5
0	NP Vst	NP			
1		NP VP			
2			P V		
3				Det	
4					N

S -> NP VP
S -> Vst NP
S -> S PP

VP -> V NP
VP -> VP PP

NP -> Det N
NP -> NP PP
NP -> NP NP

PP -> P NP

ANALYSIS (4)

	time 1	flies 2	like 3	an 4	arrow 5
0	NP Vst	NP S			
1		NP VP			
2			P V		
3				Det	
4					N

S -> NP VP
S -> Vst NP
S -> S PP

VP -> V NP
VP -> VP PP

NP -> Det N
NP -> NP PP
NP -> NP NP

PP -> P NP

ANALYSIS (5)

	time 1	flies 2	like 3	an 4	arrow 5
0	NP Vst	NP S S			
1		NP VP			
2			P V		
3				Det	
4					N

S -> NP VP
S -> Vst NP
S -> S PP

VP -> V NP
VP -> VP PP

NP -> Det N
NP -> NP PP
NP -> NP NP

PP -> P NP

ANALYSIS (6)

	time 1 flies	2 like	3 an	4 arrow	5
0	NP Vst	NP S S			
1		NP VP	–	–	
2			P V	–	
3				Det	
4					N

S -> NP VP
S -> Vst NP
S -> S PP

VP -> V NP
VP -> VP PP

NP -> Det N
NP -> NP PP
NP -> NP NP

PP -> P NP

ANALYSIS (7)

	time 1	flies 2	like 3	an 4	arrow 5
0	NP Vst	NP S S	–	–	
1		NP VP	–	–	
2			P V	–	
3				Det	NP
4					N

S -> NP VP
S -> Vst NP
S -> S PP

VP -> V NP
VP -> VP PP

NP -> Det N
NP -> NP PP
NP -> NP NP

PP -> P NP

ANALYSIS (8)

	time 1 flies	2 like	3 an	4 arrow	5
0	NP Vst	NP S S			
1		NP VP	–		
2			P V	–	
3				Det	NP
4					N

S -> NP VP
S -> Vst NP
S -> S PP

VP -> V NP
VP -> VP PP

NP -> Det N
NP -> NP PP
NP -> NP NP

PP -> P NP

ANALYSIS (9)

	time 1	flies 2	like 3	an 4	arrow 5
0	NP Vst	NP S S	–		
1		NP VP	–	–	
2			P V	–	PP
3				Det	NP
4					N

S -> NP VP
S -> Vst NP
S -> S PP

VP -> V NP
VP -> VP PP

NP -> Det N
NP -> NP PP
NP -> NP NP

PP -> P NP

ANALYSIS (10)

	time 1 flies	2 like	3 an	4 arrow	5
0	NP Vst	NP S S	–		
1		NP VP	–	–	
2			P V	–	PP VP
3				Det	NP
4					N

S -> NP VP
S -> Vst NP
S -> S PP
VP -> V NP
VP -> VP PP
NP -> Det N
NP -> NP PP
NP -> NP NP
PP -> P NP

ANALYSIS (11)

time	1	flies	2	like	3	an	4	arrow	5
0	NP Vst		NP S S		–				
1			NP VP		–		–		
2					P V		–		PP VP
3							Det		NP
4									N

S -> NP VP
S -> Vst NP
S -> S PP

VP -> V NP
VP -> VP PP

NP -> Det N
NP -> NP PP
NP -> NP NP

PP -> P NP

ANALYSIS (12)

time	1 flies	2 like	3 an	4 arrow	5
0	NP Vst	NP S S	–	–	
1		NP VP	–	–	NP
2			P V	–	PP VP
3				Det	NP
4					N

S -> NP VP
S -> Vst NP
S -> S PP

VP -> V NP
VP -> VP PP

NP -> Det N
NP -> NP PP
NP -> NP NP

PP -> P NP

ANALYSIS (13)

time 1 flies 2 like 3 an 4 arrow 5					
0	NP Vst	NP S S	–	–	
1		NP VP	–	–	NP S
2			P V	–	PP VP
3				Det	NP
4					N

S -> NP VP
S -> Vst NP
S -> S PP

VP -> V NP
VP -> VP PP

NP -> Det N
NP -> NP PP
NP -> NP NP

PP -> P NP

ANALYSIS (14)

time 1 flies 2 like 3 an 4 arrow 5					
0	NP Vst	NP S S	–	–	
1		NP VP	–	–	NP S VP
2			P V	–	PP VP
3				Det	NP
4					N

S -> NP VP
S -> Vst NP
S -> S PP

VP -> V NP
VP -> VP PP

NP -> Det N
NP -> NP PP
NP -> NP NP

PP -> P NP

ANALYSIS (15)

	time 1	flies 2	like 3	an 4	arrow 5
0	NP Vst	NP S S	–	–	
1		NP VP	–	–	NP S VP
2			P V	–	PP VP
3				Det	NP
4					N

S -> NP VP
S -> Vst NP
S -> S PP

VP -> V NP
VP -> VP PP

NP -> Det N
NP -> NP PP
NP -> NP NP

PP -> P NP

ANALYSIS (16)

	time 1	flies 2	like 3	an 4	arrow 5
0	NP Vst	NP S S	–	–	NP
1		NP VP	–	–	NP S VP
2			P V	–	PP VP
3				Det	NP
4					N

S -> NP VP
S -> Vst NP
S -> S PP

VP -> V NP
VP -> VP PP

NP -> Det N
NP -> NP PP
NP -> NP NP

PP -> P NP

ANALYSIS (17)

	time 1	flies 2	like 3	an 4	arrow 5
0	NP Vst	NP S S	–	–	NP S
1		NP VP	–	–	NP S VP
2			P V	–	PP VP
3				Det	NP
4					N

S -> NP VP
S -> Vst NP
S -> S PP

VP -> V NP
VP -> VP PP

NP -> Det N
NP -> NP PP
NP -> NP NP

PP -> P NP

ANALYSIS (18)

	time 1	flies 2	like 3	an 4	arrow 5
0	NP Vst	NP S S	–	–	NP S S
1		NP VP	–	–	NP S VP
2			P V	–	PP VP
3				Det	NP
4					N

S -> NP VP
S -> Vst NP
S -> S PP

VP -> V NP
VP -> VP PP

NP -> Det N
NP -> NP PP
NP -> NP NP

PP -> P NP

ANALYSIS (19)

	time 1	flies 2	like 3	an 4	arrow 5
0	NP Vst	NP S S	–	–	NP S S
1		NP VP	–	–	NP S VP
2			P V	–	PP VP
3				Det	NP
4					N

S -> NP VP
S -> Vst NP
S -> S PP

VP -> V NP
VP -> VP PP

NP -> Det N
NP -> NP PP
NP -> NP NP

PP -> P NP

ANALYSIS (20)

	time 1	flies 2	like 3	an 4	arrow 5
0	NP Vst	NP S S	–	–	NP S S NP
1		NP VP	–	–	NP S VP
2			P V	–	PP VP
3				Det	NP
4					N

S -> NP VP
S -> Vst NP
S -> S PP

VP -> V NP
VP -> VP PP

NP -> Det N
NP -> NP PP
NP -> NP NP

PP -> P NP

ANALYSIS (21)

	time 1	flies 2	like 3	an 4	arrow 5
0	NP Vst	NP S S	–	–	NP S S NP S
1		NP VP	–	–	NP S VP
2			P V	–	PP VP
3				Det	NP
4					N

S -> NP VP
S -> Vst NP
S -> S PP

VP -> V NP
VP -> VP PP

NP -> Det N
NP -> NP PP
NP -> NP NP

PP -> P NP

ANALYSIS (22)

	time 1	flies 2	like 3	an 4	arrow 5
0	NP Vst	NP S S	–	–	NP S S NP S S
1		NP VP	–	–	NP S VP
2			P V	–	PP VP
3				Det	NP
4					N

S -> NP VP
S -> Vst NP
S -> S PP

VP -> V NP
VP -> VP PP

NP -> Det N
NP -> NP PP
NP -> NP NP

PP -> P NP

ANALYSIS (23)

	time 1	flies 2	like 3	an 4	arrow 5
0	NP Vst	NP S S	–	–	NP S S NP S S S
1		NP VP	–	–	NP S VP
2			P V	–	PP VP
3				Det	NP
4					N

S -> NP VP
S -> Vst NP
S -> S PP

VP -> V NP
VP -> VP PP

NP -> Det N
NP -> NP PP
NP -> NP NP

PP -> P NP

BUILDING ONE SELECTED RESULT (1)

	time 1	flies 2	like 3	an 4	arrow 5
0	NP Vst	NP S S	–	–	NP S S NP S S S
1		NP VP	–	–	NP S VP
2			P V	–	PP VP
3				Det	NP
4					N

S

S -> NP VP
S -> Vst NP
S -> S PP

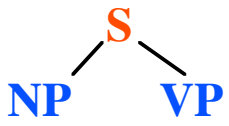
VP -> V NP
VP -> VP PP

NP -> Det N
NP -> NP PP
NP -> NP NP

PP -> P NP

BUILDING ONE SELECTED RESULT (2)

	time 1	flies 2	like 3	an 4	arrow 5
0	NP Vst	NP S S	–	–	NP S S NP S S S
1		NP VP	–	–	NP S VP
2			P V	–	PP VP
3				Det	NP
4					N



S -> NP VP
S -> Vst NP
S -> S PP

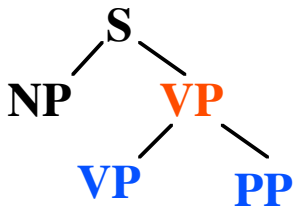
VP -> V NP
VP -> VP PP

NP -> Det N
NP -> NP PP
NP -> NP NP

PP -> P NP

BUILDING ONE SELECTED RESULT (3)

	time 1 flies	2 like	3 an	4 arrow	5
0	NP Vst	NP S S	–	–	NP S S NP S S S
1		NP VP	–	–	NP S VP
2			P V	–	PP VP
3				Det	NP
4					N



S -> NP VP
S -> Vst NP
S -> S PP

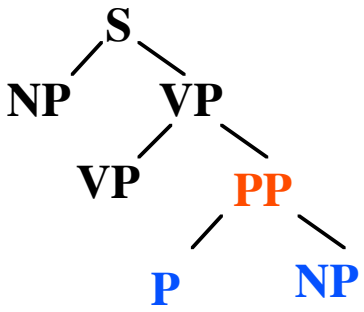
VP -> V NP
VP -> VP PP

NP -> Det N
NP -> NP PP
NP -> NP NP

PP -> P NP

BUILDING ONE SELECTED RESULT (4)

	time 1	flies 2	like 3	an 4	arrow 5
0	NP Vst	NP S S	–	–	NP S S NP S S S
1		NP VP	–	–	NP S VP
2			P V	–	PP VP
3				Det	NP
4					N



S -> NP VP
S -> Vst NP
S -> S PP

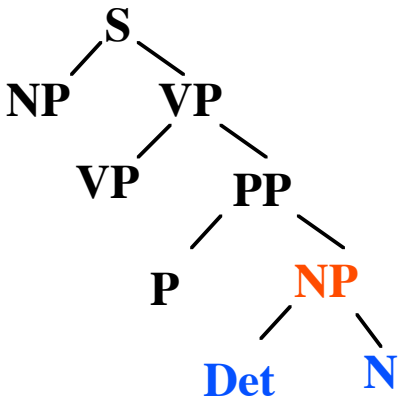
VP -> V NP
VP -> VP PP

NP -> Det N
NP -> NP PP
NP -> NP NP

PP -> P NP

BUILDING ONE SELECTED RESULT (5)

	time 1	flies 2	like 3	an 4	arrow 5
0	NP Vst	NP S S	–	–	NP S S NP S S S
1		NP VP	–	–	NP S VP
2			P V	–	PP VP
3				Det	NP
4					N



S -> NP VP
S -> Vst NP
S -> S PP

VP -> V NP
VP -> VP PP

NP -> Det N
NP -> NP PP
NP -> NP NP

PP -> P NP

LIMITATIONS OF THE CYK ALGORITHM

Built-in assumptions

Chomsky Normal Form grammars

Breadth-first (exhaustive): always compute all values for each cell

Rigid control structure: bottom-up, left-to-right (one diagonal at a time)

Improvements

Chart as a data structure with more flexible computation

Also used for generation