

Natuurlijke Taalverwerking

Natural Language Processing

Gosse Bouma and Begoña Villada

3e trimester 2002/2003

Chart Parsing

1. Disadvantages of Backtracking,
2. Don't compute things twice,
3. Chart,
4. Prolog Implementation,
5. Partial Parsing.

Depth-first search

- If there is a **choice** (between rules, between shift or reduce action),
- explore one option completely (**depth-first**),
- and other options only if first choice fails.
- often implemented using **back-tracking**.

Depth-first parsing

- DCG: **top-down**, depth-first
- Shift-reduce: **bottom-up**, depth-first

Disadvantages of backtracking

- De spectaculaire groei van het Amsterdamse bureau Franzen , Hey & Veltman (FHV) heeft zich ook vorig jaar voortgezet.

SR parsing with Backtracking

	String	Stack	action	rule
1	van het Amsterdamse ..	[det,a,n]	red	
2	van het Amsterdamse ..	[det,n]	red	
3	van het Amsterdamse ..	[np]	sh	
	...			
n	heeft zich ook	[np,pp]	sh	
	...	(fail)		
2'	van het Amsterdamse ..	[det,n]	sh	
n'	heeft zich ook	[det,n,pp]	red	
	..			

- PP **van ... (FHV)** derived twice

Causes of backtracking

- Ambiguity,
- Need to find **all solutions**,
- Ungrammatical input.

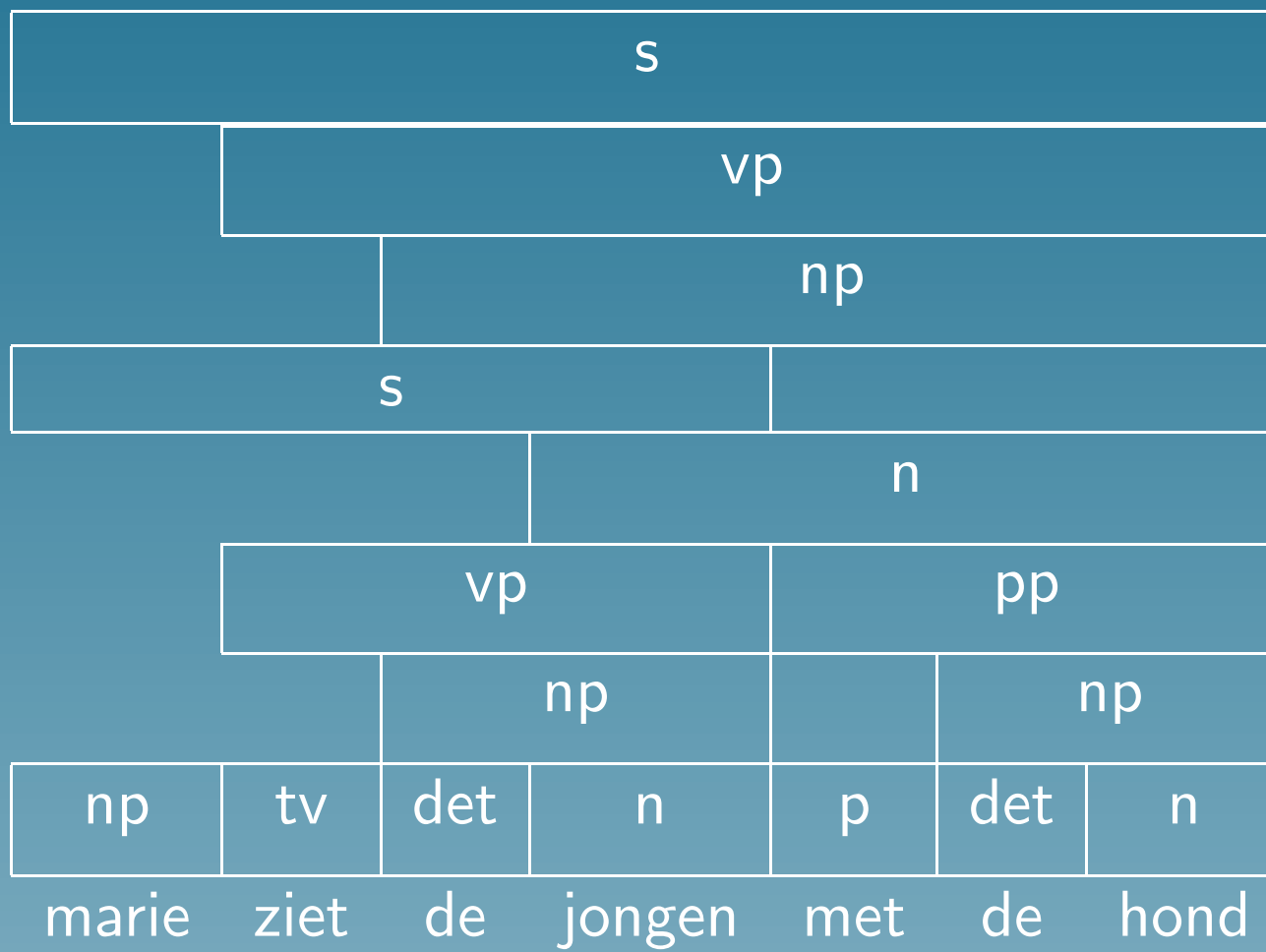
Partial Parsing

- Find all NPs (or other category) in a given input,
- Implementation with shift-reduce parser:
 - ★ Parse all substrings of the input,
 - ★ Input of length 20 has 20! substrings
 - ★ Many NPs will be recognized several times....

Chart Parsing

- Avoid computing anything twice!
- Store intermediate results in well-formed substring table (chart):
 - ★ List of constituents found in the input

A Chart



Representing a Chart

0 Marie 1 ziet 2 de 3 jongen 4 met 5 de 6 hond 7

$\langle 0,1,np \rangle$ $\langle 1,2,tv \rangle$ $\langle 2,3,det \rangle$

$\langle 3,4,n \rangle$ $\langle 2,4,np \rangle$ $\langle 1,4,vp \rangle$

$\langle 0,4,s \rangle$ $\langle 4,5,p \rangle$ $\langle 5,6,det \rangle$

$\langle 6,7,n \rangle$ $\langle 5,7,np \rangle$ $\langle 4,7,pp \rangle$

$\langle 1,7,vp \rangle$ $\langle 3,7,np \rangle$ $\langle 0,7,s \rangle$

Chart Parsing

- **Goal:** find all constituents in the input,
- **Success:** there is a constituent spanning the input of category S .

Chart Parsing

- **Rule 1:** Add item $\langle i, i + 1, Cat \rangle$ if there is a **word** W at position i with lexical category Cat .
- **Rule 2:** Add item $\langle i, n, Cat \rangle$ if there is a **rule** $Cat \rightarrow C_1 \dots C_n$ and **items** $\langle i, j, C_1 \rangle \dots \langle m, n, C_n \rangle$

Implementation in Prolog

- Chart: collection of facts of the form **constituent (Begin, End, Category)**.
- Adding items to the chart: use **assert**.
- Process the string from left to right,
- At each position: add **all** possible items to the chart (once) (use **failure-driven loop**).

Partial Parsing

- Find all NPs in a string,
- No full parse required.
- Easy using a chart,
- Hard (and slow) using shift-reduce parser

Partial Parsing Experiment

	Runtime (ms)	
Sentences	Shift-Reduce	Chart
5	804	5
10	1229	12
20	41295	26

Exercise 4

- Use an existing chart-parser implementation to do partial parsing,
- Write predicates to present parse results to the user,
- Work with a corpus of text tagged with Part of Speech info.

Exercise 4: Example

```
zin(13, [art(bep, zijd_of_mv, neut, 'De'),  
        adj(attr, stell, verv_neut, enige),  
        n(soort, ev, neut, maatregel),  
        ...
```

```
13 [De enige maatregel] die [de Duitsers]  
mogen nemen om hun boeren te beschermen  
tegen [de gevolgen van de koersstijging van  
de mark] is [een concrete sluiting van de  
grenzen] voor buitenlandse granen en  
zuivelprodukten .
```

Exam

- Friday, June, 27, 9-12, Heymanszaal, Academieggebouw
- Example Questions on the Web.