

## Sprachorientierte KI: Syntax und Parsing

- Syntax als Untersuchungsgegenstand
- Wortartendisambiguierung
- Phrasenstrukturgrammatiken
- Parsing mit Phrasenstrukturgrammatiken
- Restrikingierte Phrasenstrukturgrammatiken
- Unifikationsgrammatiken
- Constraint-basierte Grammatiken
- Robustes Parsing



## Parsing mit Phrasenstrukturgrammatiken

- Strukturanalyse
- flaches Parsing
- CFG-Parsing
- Parsingstrategien
- Tabellenparsing
- Chartparsing
- Deterministisches Parsing
- Stochastisches Parsing



## Strukturanalyse

- Parsing natürlicher Sprache
  - partes orationes: Grammatische Analyse als Wortartenbestimmung
  - Zerlegung eines Satzes in seine Bestandteile
  - Strukturaufklärung, Ermittlung der grammatischen Relationen zwischen den Satzbestandteilen
- Ziel: Strukturbeschreibungen
  - Grundlage für eine semantische Interpretation
  - Erkenner vs. Parser



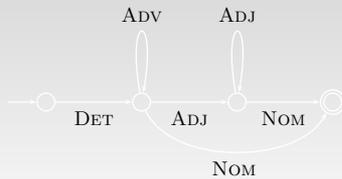
## Strukturanalyse

- Ergebnisstrukturen
  - syntaktische Phrasenstrukturbäume
  - syntaktische Dependenzrelationen
  - semantische Repräsentationen



## Flaches Parsing

- Chunking: Zerlegung eines Satzes in Konstituenten ohne Beachtung der rekursiven Einbettung
- z.B. mit gewichteten endlichen Automaten



## CFG-Parsing

Lexikon	Grammatik
N → Marie	S → NP VP
N → Hans	VP → V PP
N → Park	VP → V PP PP
P → auf	PP → P NP
P → im	NP → N
V → wartet	



## CFG-Parsing

- Regelanwendung von links nach rechts: top down
  - Ableitung des Satzes aus dem Startsymbol:
    - S
    - NP VP
    - N V PP
    - Hans wartet P NP
    - ...
- Regelanwendung von rechts nach links: bottom up
  - Ableitung des Startsymbols aus dem Satz:
    - Hans wartet auf Marie
    - N V P N
    - NP V P NP
    - NP V PP
    - ...



## CFG-Parsing

- alle Alternativen für Regelanwendungen müssen berücksichtigt werden
- Mehrdeutigkeiten (Ambiguitäten)
  - verhindern lokale Entscheidungen
- lexikalische Mehrdeutigkeiten:
  - laufen/VINF/VFIN/NN, schöne/ADJ/NN ...
- strukturelle als Folge lexikalischer Mehrdeutigkeit
  - ... dort, wo die wilden/ADJ/NN tiere jagen/V<sub>intr</sub>/V<sub>tr</sub>



## CFG-Parsing

- rein strukturelle Mehrdeutigkeit  
 $[NP\ der\ Mann\ [PP\ mit\ dem\ Hut\ [PP\ auf\ der\ Stange]]]$   
 $[NP\ der\ Mann\ [PP\ mit\ dem\ Hut]\ [PP\ auf\ der\ Stange]]$   
..., weil  $[NP\ dem\ Sohn\ des\ Meisters]\ [NP\ Geld]\ fehlt.$   
..., weil  $[NP\ dem\ Sohn]\ [NP\ des\ Meisters\ Geld]\ fehlt.$
- lokale Mehrdeutigkeiten
  - können im weiteren Verlauf der Analyse aufgelöst werden
- globale Mehrdeutigkeiten
  - bleiben bis zum Abschluss der Analyse bestehen



## CFG-Parsing

- Parsing als Suche
  - alternative Regelanwendungen spannen einen Suchraum auf



## CFG Parsing

- kontextfreie Phrasenstrukturregeln werden möglicherweise erst im Parser generiert
- ID/LP-Grammatiken  
 $S \rightarrow \{NP, VP\}$      $NP < VP$ 
  - LP-Constraints meist auf der Basis von Merkmalen für komplexe Kategorien formuliert



## CFG Parsing

- Dependenzgrammatiken
  - Valenzmuster  
*füttern*:  $\langle SUBJ, *, DOBJ \rangle$
  - "flache" Phrasenstrukturregel  
 $S \rightarrow C1_{SUBJ} VFIN C2_{DOBJ}$
  - "tiefe" Phrasenstrukturregeln  
 $S \rightarrow C1_{SUBJ} C2_{VP}$   
 $C2_{VP} \rightarrow VFIN C3_{DOBJ}$



## Parsingstrategien

erwartungsgesteuert (top-down, expand-reduce)

- Problem: links-/rechtsrekursive Regeln verursachen Terminierungsprobleme
  - auch über mehrere Regeln hinweg:  
 $X \rightarrow Y a$   
 $Y \rightarrow X$
- Lösung: Transformation in schwach äquivalente Grammatik ohne Links-/Rechtsrekursion
  - linguistisch motivierte Ableitungsstruktur geht verloren
  - Ausweg: separater Strukturaufbau durch Unifikation



## Parsingstrategien

datengesteuert (bottom-up, shift-reduce)

- Problem: leere Produktionen  $X \rightarrow \epsilon$ 
  - eventuell "Lizensierung" durch lexikalische Knoten
- Problem: unäre Regeln, die Zyklen bilden
  - vermeiden



## Parsingstrategien

Tiefe zuerst

- alternative Regelanwendungen werden erst später untersucht
- Speicherung auf einem Stack

Breite zuerst

- alternative Regelanwendungen werden "gleichzeitig" abgearbeitet
- Verwaltung der Alternativen in einer Warteschlange



## Parsingstrategien

links-rechts

- Symbolfolgen werden von links beginnend abgearbeitet

rechts-links

- Symbolfolgen werden von rechts beginnend abgearbeitet



## Parsingstrategien

- gemischte Strategien
  - Left-Corner-Parsing: Top-down-Analyse mit Aktivierung der Regeln durch die linke Ecke
  - Robustheit gegenüber fehlerhaftem Input: bottom up-Analyse mit top down-Rekonstruktion im Fehlerfall (MELLISH 1989)
  - Inselparsing: bidirektionale Analyse von sicheren Hypothesen ausgehend (z.B. Spracherkennung)



## Tabellenparsing

- Effizienzproblem: Mehrfachanalysen in unterschiedlichen Analysepfaden
- Daten
  - Deutsch mit kopffinaler Verbgruppe
  - Normalfall: Nebensatzreihung
    - ..., weil der Vater seine Kinder liebt.
    - ..., weil der Vater seinen Kindern glaubt.
    - ..., weil der Vater seinen Kindern ein Eis verspricht.
    - ..., weil der Vater seinen Kindern mit einer Strafe droht.

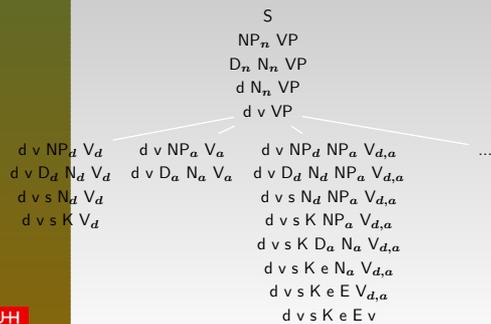


## Tabellenparsing

- Grammatik
  - $S' \rightarrow \text{Konj } S$
  - $S \rightarrow NP_n VP$
  - $VP \rightarrow NP_a V_a$
  - $VP \rightarrow NP_d V_d$
  - $VP \rightarrow NP_d NP_a V_{d,a}$
  - $VP \rightarrow NP_d PP_{mit,d} V_{d,mit}$
  - $NP_x \rightarrow D_x N_x$
  - $PP_{x,y} \rightarrow P_x NP_y$
- Beispielanalyse: top-down, Tiefe-zuerst
  - ... der Vater seinen Kindern ein Eis verspricht.



## Tabellenparsing

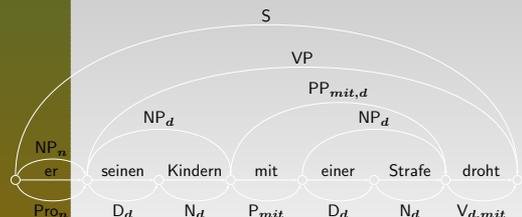


## Tabellenparsing

- Well-formed Substring Table (Chart)
  - gerichteter azyklischer Graph mit
    - einer Quelle (Satzanfang)
    - einer Senke (Satzende) und
    - einer totalen Präzedenzrelation über den Knoten
  - Kanten entsprechen erfolgreich erkannten Konstituenten



## Tabellenparsing



## Tabellenparsing

	1	2	3	4	5	6	7
0	er Pro <sub>n</sub> NP <sub>n</sub>						S
1		seinen D <sub>d</sub>	NP <sub>d</sub>				VP
2			Kindern N <sub>d</sub>				
3				mit P <sub>mit</sub>		PP <sub>mit</sub>	
4					einer D <sub>d</sub>	NP <sub>d</sub>	
5						Strafe N <sub>d</sub>	
6							droht V <sub>d,mit</sub>



## Tabellenparsing

- Cocke-Younger-Kasami-Algorithmus (KASAMI 1965, YOUNGER 1967)
- Grammatik in Chomsky-Normalform
  - binärverzweigende Regeln:  $X \rightarrow YZ$
  - präterminale Regeln:  $X \rightarrow a$



## Tabellenparsing

- CYK: Eigenschaften
  1. Länge der Ableitung konstant:
    - $n$  lexikalische Regeln +  $n-1$  syntaktische Regeln
  2. Anzahl der Binärzerlegungen eines Satzes ist konstant:  $n-1$ 
    - $((a) (b c d))$
    - $((a b) (c d))$
    - $((a b c) (d))$
  3. Strukturelle Mehrdeutigkeiten durch unterschiedliche Überdeckung der Eingabekette entfallen
    - $VP \rightarrow NP NP V$
    - $VP \rightarrow NP V$
    - $VP \rightarrow V$

## Tabellenparsing

- CYK: Algorithmus
  1. Initialisieren der Matrix
    - für  $i = 0$  bis  $n - 1$ :
    - $CHART_{i,i+1} \leftarrow \{ X \mid X \in V_T \text{ und } w_{i+1} \in X \}$
  2. Berechnung der übrigen Felder
    - für  $k = 2$  bis  $n$ :
    - für  $i = 0$  bis  $n - k$ :
    - $j \leftarrow i + k$
    - $CHART_{i,j} \leftarrow \{ A \mid (A \rightarrow XY) \in R \wedge \exists m . (X \in CHART_{i,m} \wedge Y \in CHART_{m,j}, \text{ mit } i < m < j) \}$

wenn  $S \in CHART_{0,n}$   
dann RETURN(*true*)  
sonst RETURN(*false*)

## Tabellenparsing

- Bottom up-Analyse
  - Zeitbedarf  $\mathcal{O}(n^3)$
  - Speicherbedarf  $\mathcal{O}(n^2)$
  - durch Wiederverwendung von Zwischenergebnissen
- Nachteil: es werden immer noch Konstituenten generiert, die in keine übergreifenden Strukturen eingebaut werden können
  - EARLEY-Parser

## Chart-Parsing

- Aktive Chart
  - Erweiterung: auch Versuche einer Regelanwendung werden in die Chart eingetragen
  - Aktive Kanten: offene Erwartungen für den rechten Kontext
    - Notation:  $\langle a, b, A \rightarrow B . C D \rangle$
  - Inaktive Kanten: vollständig gesättigte Erwartungen für den rechten Kontext
    - Notation:  $\langle a, b, A \rightarrow B C D . \rangle$

## Chart-Parsing

- TD-Regel (Initialisierung)
  - Für alle Regeln  $A \rightarrow w_1$  bei denen  $A$  ein Startsymbol der Grammatik ist, füge eine Kante  $\langle 0, 0, A \rightarrow . w_1 \rangle$  in die Chart ein.
- Regel:  $S \rightarrow NP_n VP$



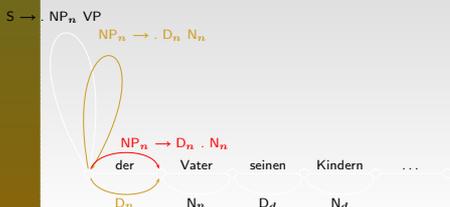
## Chart-Parsing

- TD-Regel (Kanteneinführung)
  - Beim Eintragen einer Kante  $\langle i, j, A \rightarrow w_1 . B w_2 \rangle$  ergänze für jede Regel  $B \rightarrow w_3$  eine Kante  $\langle j, j, B \rightarrow . w_3 \rangle$ .
- Regel:  $NP_X \rightarrow D_X N_X$



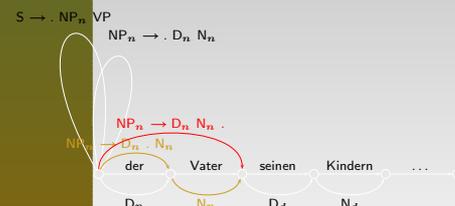
## Chart-Parsing

- Fundamentalregel (Kantenerweiterung)
  - Enthält die Chart zwei Kanten  $\langle i, j, A \rightarrow w_1 . B w_2 \rangle$  und  $\langle j, k, B \rightarrow w_3 . \rangle$ , dann füge eine dritte Kante  $\langle i, k, A \rightarrow w_1 B . w_2 \rangle$  hinzu.



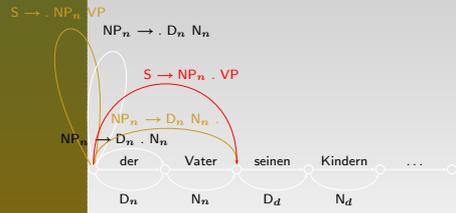
## Chart-Parsing

- Erneute Anwendung der Fundamentalregel



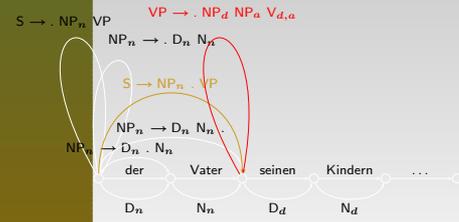
## Chart-Parsing

### Erneute Anwendung der Fundamentalregel



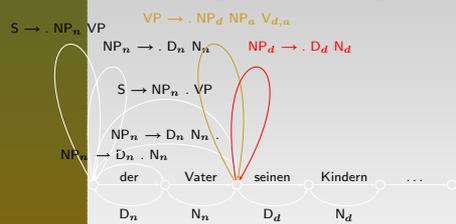
## Chart-Parsing

### Erneute Anwendung der Top-Down-Regel



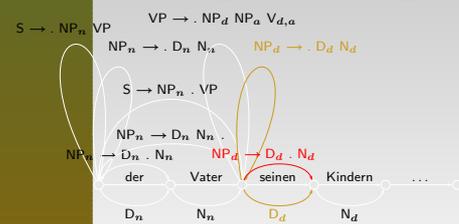
## Chart-Parsing

### Erneute Anwendung der Top-Down-Regel



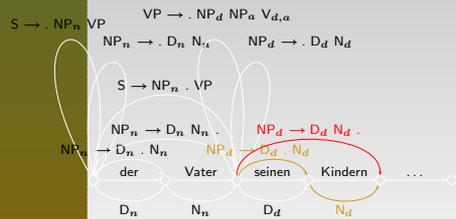
## Chart-Parsing

### Erneute Anwendung der Fundamentalregel



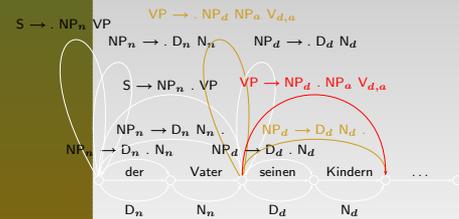
## Chart-Parsing

### Erneute Anwendung der Fundamentalregel



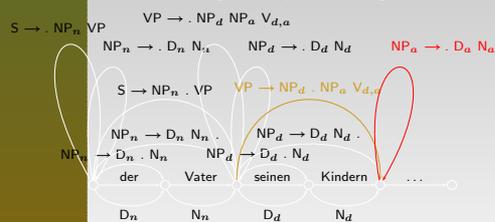
## Chart-Parsing

### Erneute Anwendung der Fundamentalregel



## Chart-Parsing

### Erneute Anwendung der Top-Down-Regel



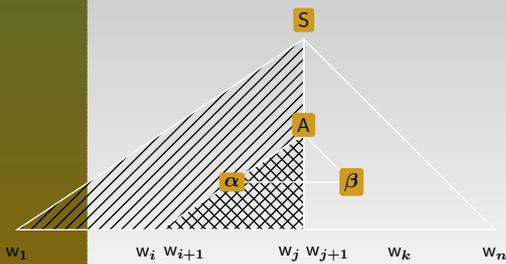
## Chart-Parsing

### Earley-Algorithmus (EARLEY 1970)

- für beliebige CF-Grammatiken
  - auch mit Rekursion, Zyklen und Tilgungen
  - gemischte Top down/Bottom up-Strategie, um die Erzeugung nicht weiter verwendbarer Konstituenten weitgehend zu vermeiden
1. Top-Down-Bedingung: es werden nur solche Kanten generiert, deren linker Kontext mit der Grammatik verträglich ist
  2. Bottom up-Bedingung: der bereits abgearbeitete Regelteil muß sich auf die Daten ableiten lassen



## Chart-Parsing



## Chart-Parsing

- Operationen
  - Expand (Top-down-Regel, Kantenintroduktion)
  - Complete (Fundamentalregel, Kantenexpansion)
  - Shift (Einbeziehung lexikalischer Kanten)
- verschiedene Suchstrategien (Tiefe/Breite) sind möglich in Abhängigkeit von der Agenda-Verwaltung

## Chart-Parsing

- EARLEY-Algorithmus
  1. Initialisierung
    - für alle  $(S \rightarrow \beta) \in R$ :  $CHART_{0,0} \leftarrow \langle S, \emptyset, \beta \rangle$
    - wende EXPAND solange auf die zuvor erzeugten Kanten an, bis keine neuen Kanten mehr erzeugt werden können.
  2. Berechnung der übrigen Kanten
    - für  $j = 1, \dots, n$ :
    - für  $i = 0, \dots, j$ :
    - berechne  $CHART_{i,j}$ :
      1. wende SHIFT auf alle geeigneten Kanten in  $CHART_{i,j-1}$  an
      2. Wende EXPAND und COMPLETE solange an, bis keine neuen Kanten mehr erzeugt werden können.

wenn  $(S, \beta, \emptyset) \in CHART_{0,n}$   
dann RETURN(true) sonst RETURN(false)

## Chart-Parsing

- Chart-basierte Algorithmen sind nur Erkennen
- Erweiterung zum Parser:
  - Extraktion von Strukturbäumen (Ableitungen) aus der Chart in einem separaten Schritt
  - Grundlage: Verwaltung eines Verweises auf die verursachende Kante in der Fundamentalregel
  - "Aufsammeln" der Bäume beginnend bei allen vollständigen S-Kanten

## Chart-Parsing

- Ressourcenbedarf: Zeit
  - $\mathcal{O}(n^3 \cdot |G^2|)$
  - für deterministische Grammatiken:  $\mathcal{O}(n^2)$
  - in vielen praktisch relevanten Fällen:  $\mathcal{O}(n)$
- Zeitkomplexität nur für das Aufbauen der Chart
- Resultatsextraktion kann bei exponentiell vielen Resultaten exponentiellen Aufwand erfordern

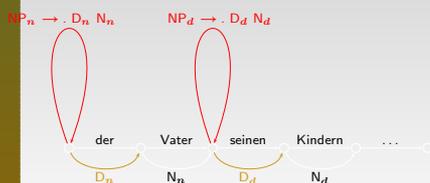
## Chart-Parsing

- Ressourcenbedarf: Speicher
  - $\mathcal{O}(n^2)$
  - durch Wiederverwendung von Zwischenergebnissen
    - nur für atomare Nichtterminalsymbole möglich
- Chart ist allgemeine Datenstruktur zur Verwaltung von Zwischenergebnissen beim Parsing
  - alternative Analysestrategien sind möglich
  - z.B. bottom-up

## Chart-Parsing

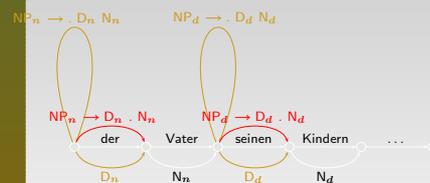
- Bottom-Up-Regel (Kanteneinführung)
 

Beim Eintragen einer Kante  $\langle i, j, B \rightarrow w_1 \rangle$  ergänze für jede Regel  $A \rightarrow B w_2$  eine Kante  $\langle i, i, A \rightarrow \cdot B w_1 \rangle$



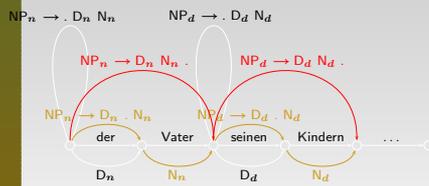
## Chart-Parsing

- Anwendung der Fundamentalregel



## Chart-Parsing

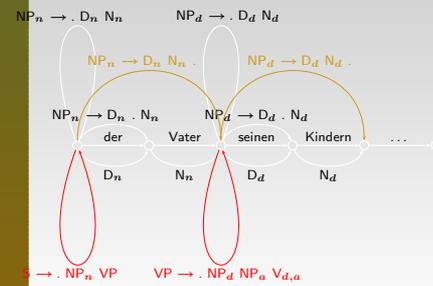
### Anwendung der Fundamentalregel



Wolfgang Menzel: Sprachorientierte KI: Syntax und Parsing – p. 49

## Chart-Parsing

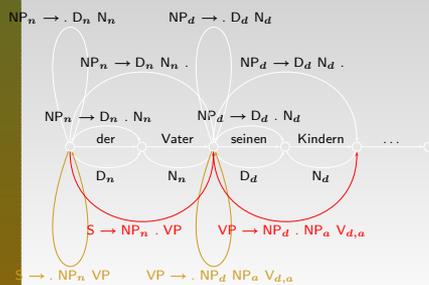
### Anwendung der Bottom-Up-Regel



Wolfgang Menzel: Sprachorientierte KI: Syntax und Parsing – p. 50

## Chart-Parsing

### Anwendung der Fundamentalregel



Wolfgang Menzel: Sprachorientierte KI: Syntax und Parsing – p. 51

## Chart-Parsing

### Chart wächst monoton

- Kanten werden nicht entfernt
  - auch gescheiterte Regelanwendungen werden aufbewahrt
  - nicht mehr expansionsfähige aktive Kanten
- Mehrfacharbeit wird vermieden
- Kante wird nur dann in die Chart eingetragen, wenn dort noch nicht vorhanden

Wolfgang Menzel: Sprachorientierte KI: Syntax und Parsing – p. 52

## Chart-Parsing

### Agenda

- Liste der aktiven Kanten
- beliebig sortierbar
- Kellerspeicher: Tiefe-zuerst
- Warteschlange: Breite zuerst
- TD-Regel: erwartungsgesteuerte Analyse
- BU-Regel: datengesteuerte Analyse

Wolfgang Menzel: Sprachorientierte KI: Syntax und Parsing – p. 53

## Chart-Parsing

### flexible Steuerung für Mischstrategien

- left-corner-Parsing
  - TD-Parsing, aber nur diejenigen Regeln aktivieren, die eine gegebene lexikalische Kategorie (linke Ecke) direkt oder indirekt ableiten können
  - Tabelle mit der Zuordnung zwischen Regeln und ihren möglichen linken Ecken wird aus der Grammatik berechnet
  - Variante: head-corner Parsing

Wolfgang Menzel: Sprachorientierte KI: Syntax und Parsing – p. 54

## Chart-Parsing

### best-first-Parsing

- Sortieren der Agenda nach Konfidenzwerten
- Hypothesenbewertungen der Spracherkennung
- Regelbewertungen (z.B. Häufigkeit in Korpus)

Wolfgang Menzel: Sprachorientierte KI: Syntax und Parsing – p. 55

## Chart-Parsing

### Aufgabe der totalen Präzedenzrelation über den Knoten:

- Lattice-Parsing
  - Worthypothesegraphen bei der Verarbeitung gesprochener Sprache
  - z.B. Ergebnisse einer HMM-Worterkennung
- Aufgabe der Verbundenheit des Hypothesegraphen:
  - Grid-Parsing
    - z.B. Ergebnisse eines Wordspotters mit Hypothesealternativen

Wolfgang Menzel: Sprachorientierte KI: Syntax und Parsing – p. 56

## Deterministisches Parsing

- nur für Teilmengen der CF-Sprachen: LR(n),  $n=0, 1, \dots, k$  deterministische Sprachen
- Berechnung einer Parsingtabelle aus der Grammatik vor Beginn der Analyse
- *eindeutige* Ermittlung der nächsten Aktion:

shift	lege das nächste Wort auf den Stack
reduce	reduziere den Stack
accept	Satz wurde akzeptiert
error	Satz wurde zurückgewiesen, nicht in L(G)



## Deterministisches Parsing

- Tomita-Parsing (TOMITA 1984)
- für beliebige CF-Grammatiken (ohne Zyklen)
- zwei Erweiterungen:
  1. Parsingtabelle mit alternativen Einträgen  
→ baumstrukturierter Stack
  2. Zusammenfassen alternativer Analysevarianten  
→ graphstrukturierter Stack

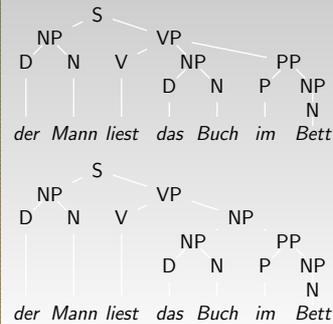


## Deterministisches Parsing

- Tomita-Parsing
- effiziente Strukturverwaltung (packed shared forests)
  1. Zusammenfassen gemeinsamer Teilstrukturen verschiedener Syntaxbäume  
→ subtree sharing
  2. Zusammenfassen von Teilstrukturen mit gleichem Spitzenknoten und gleichen Terminalknoten  
→ local ambiguity packing

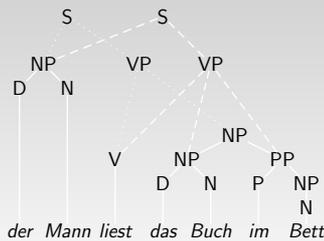


## Deterministisches Parsing



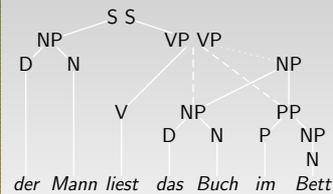
## Deterministisches Parsing

- subtree-sharing



## Deterministisches Parsing

- subtree-sharing und local ambiguity packing



- Zeitaufwand:  $O(n^3 \cdot |G|^2)$  (KIPPS 1991)



## Stochastisches Parsing

- Warum stochastisches Parsing?
- Stochastisches Basismodell
- Evaluation
- Training
- Parsing
- Erweitertes Basismodell
- Alternative Modelle



## Warum stochastisches Parsing?

- gemeinsame Probleme aller symbolischen Parser:
  - hohe Ergebnismehrdeutigkeit
  - auch bei (sehr) feiner syntaktischer Modellierung
  - trotz geringer Abdeckung
- Ergebnismehrdeutigkeit und Abdeckungsgrad sind typischerweise gegenläufig



## Warum stochastisches Parsing?

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



## Warum stochastisches Parsing?

- Ergebnismehrdeutigkeit
  - Hinter dem Betrug werden die gleichen Täter vermutet, die während der vergangenen Tage in Griechenland gefälschte Banknoten in Umlauf brachten.
  - Paragram (KUHN UND ROHRER 1997): 92 Lesarten



## Warum stochastisches Parsing?

- Ergebnismehrdeutigkeit
  - Hinter dem Betrug werden die gleichen Täter vermutet, die während der vergangenen Tage in Griechenland gefälschte Banknoten in Umlauf brachten.
  - Paragram (KUHN UND ROHRER 1997): 92 Lesarten
  - Gepard (LANGER 2001): 220 Lesarten



## Warum stochastisches Parsing?

- Ergebnismehrdeutigkeit
  - Hinter dem Betrug werden die gleichen Täter vermutet, die während der vergangenen Tage in Griechenland gefälschte Banknoten in Umlauf brachten.
  - Paragram (KUHN UND ROHRER 1997): 92 Lesarten
  - Gepard (LANGER 2001): 220 Lesarten
  - durchschnittliche Ambiguität über ein Zeitungstextkorpus: 78 bei einer mittleren Satzlänge von 11.43 syntaktischen Wörtern (Gepard)
  - Extremfall:  $6.4875 \cdot 10^{22}$  Lesarten für einen Satz (BLOCK 1995)



## Warum stochastisches Parsing?

- Ambiguitätsquellen:
  - lexikalische Mehrdeutigkeit
  - Attachment
    - We saw the Eiffel Tower flying to Paris.
  - Koordination:
    - alte Männer und Frauen
  - NP-Klammerung
    - ... der Sohn des Meisters Geld



## Warum stochastisches Parsing?

- Beispiel: PP-Attachment  
der Ball mit den Punkten in der Tasche auf dem Tisch
- wächst exponentiell (Catalan) mit der Anzahl der PPs

$$C(n) = \frac{1}{n+1} \binom{2n}{n}$$

# PPs	# Parses
2	2
3	5
4	14
5	132
6	469
7	1430
8	4867



## Warum stochastisches Parsing?

- Abdeckung (coverage)
  - partieller Parser (WAUSCHKUHN 1996): 56.5% der Sätze
  - Gepard: 33.51%
  - auf Testsuites (bessere Lexikonabdeckung, kürzere und weniger ambige Sätze) bis zu 66%



## Stochastisches Basismodell

- Ausweg: stochastische kontextfreie Grammatiken (PCFG)
- Schätzen von Ableitungswahrscheinlichkeiten für alle Regeln

$$\Pr(N \rightarrow \zeta)$$

bzw.

$$\Pr(N \rightarrow \zeta | N) \quad \text{mit} \quad \sum_{\zeta} \Pr(N \rightarrow \zeta) = 1$$

- z.B.

S → NP VP	0.8
S → Aux NP VP	0.15
S → VP	0.05



## Stochastisches Basismodell

- Sprachmodelle: Zuordnung einer Wahrscheinlichkeit zu einer Terminalkette

$$\Pr(w_{1,n}) = \sum_{t_{1,n}} \Pr(t_{1,n})$$

(mehrere Ableitungen für einen Satz)

$$= \sum_{t_{1,n}} \prod_{r_j \in t_{1,n}} \Pr(r_j)$$

- Ermittlung der wahrscheinlichsten Wortfolge



## Stochastisches Basismodell

- Disambiguierung: Ermittlung der wahrscheinlichsten Ableitung

$$t_{1,n} = \arg \max_{t_{1,n} \in T} \Pr(t_{1,n})$$

$$= \arg \max_{t_{1,n} \in T} \prod_{r_j \in t_{1,n}} \Pr(r_j)$$

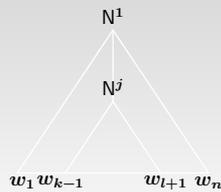


## Stochastisches Basismodell

- Unabhängigkeitsannahme:

$$\Pr(N_{k,l}^j \rightarrow \zeta | N^1, \dots, N^{j-1}, w_1, \dots, w_{k-1}, w_{l+1}, \dots, w_n)$$

$$= \Pr(N_{k,l}^j \rightarrow \zeta)$$



## Evaluation

- PARSEVAL (BLACK ET AL. 1991)
- Vergleich mit einer Referenzannotation (*gold standard*)
- labelled recall

$$LR = \frac{\# \text{ korrekte Konstituenten im Resultat}}{\# \text{ Konstituenten in der Referenz}}$$

- labelled precision

$$LP = \frac{\# \text{ korrekte Konstituenten im Resultat}}{\# \text{ Konstituenten im Resultat}}$$



## Evaluation

- crossing brackets

Ein Konstituent eines Parsebaumes enthält Teile von zwei Konstituenten aus der Referenz, ohne dass die beiden vollständig im Parseresultat enthalten sind.

Resultat: [ [ A B C ] [ D E ] ]  
Referenz: [ [ A B ] [ C D E ] ]

$$CB = \frac{\# \text{ crossing brackets}}{\# \text{ Sätze}}$$

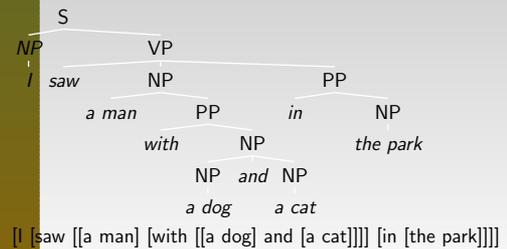
$$OCB = \frac{\# \text{ Sätze ohne crossing brackets}}{\# \text{ Sätze}}$$



## Evaluation

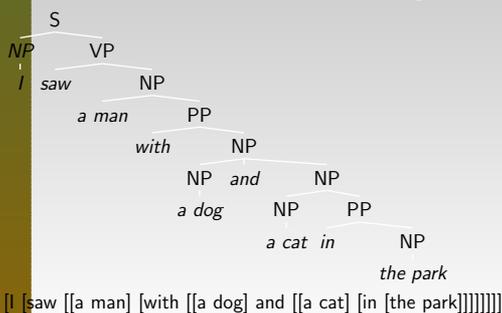
- Wie aussagekräftig sind die Qualitätsmaße?

- Referenz:



## Evaluation

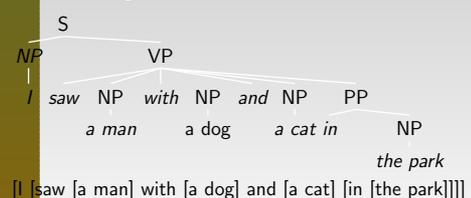
- 1. Resultat: eine fehlerhafte strukturelle Anbindung



## Evaluation

- 2. Resultat: weitgehend flache Analyse

- Vermeiden aller Entscheidungen über strukturelle Anbindungen



## Evaluation

- 1. Resultat  

$$I \text{ [saw [a man] [with [[a dog] and [[a cat] [in [the park]]]]]]]$$

$$I \text{ [saw [a man] [with [[a dog] and [a cat]]] [in [the park]]]$$

$$LR = \frac{7}{10} = 0.7 \quad LP = \frac{7}{11} = 0.64 \quad CB = \frac{3}{1} = 3$$
- 2. Resultat  

$$I \text{ [saw [a man] with [a dog] and [a cat] [in [the park]]]$$

$$I \text{ [saw [[a man] [with [[a dog] and [a cat]]]] [in [the park]]]$$

$$LR = \frac{7}{10} = 0.7 \quad LP = \frac{7}{7} = 1 \quad CB = \frac{0}{1} = 0$$
- Alternative (LIN 1996):  
 Transformation in eine Abhängigkeitsstruktur und Evaluation der Anbindungsfehler



## Training

- Schätzen der Regelanwendungswahrscheinlichkeiten
- Einfachster Fall: Treebank-Grammatiken (CHARNIAK 1996)
 
$$\Pr(N \rightarrow \zeta | N) = \frac{C(N \rightarrow \zeta)}{\sum_{\xi} C(N \rightarrow \xi)} = \frac{C(N \rightarrow \zeta)}{C(N)}$$
- Penn-Treebank: 10605 Regeln, davon 3943 nur einmal beobachtet
- Resultate für Sätze bis max. 40 Wortformen:
  - LR = 80.4%, LP = 78.8%
  - Konstituenten ohne crossing brackets: 87.7%



## Training

- Training auf unannotierten Daten: inside-outside-Algorithmus
  - Voraussetzung:
    - Grammatik (incl. Lexikon)
    - Trainingskorpus
  - Parameter-Reestimation über Ableitungsketten
  - aber: ein Satz kann üblicherweise auf verschiedenen Wegen abgeleitet werden
    - wichten der Varianten mit ihren Ableitungswahrscheinlichkeiten
    - Rekombinationsverfahren



## Training

- inside-Wahrscheinlichkeit: Wahrscheinlichkeit für Expansion eines Nichtterminals zu einer bestimmten Terminalkette
 
$$\beta_j(k, l) = \Pr(w_{k,l} | N_{k,l}^j)$$
- outside-Wahrscheinlichkeit: Wahrscheinlichkeit für die Ableitung eines Baumkontextes
 
$$\alpha_j(k, l) = \Pr(w_{1,k-1}, N_{k,l}^j, w_{l+1,n})$$



## Training

- Regelwahrscheinlichkeit (für Chomsky-Normalform)
 
$$C(N^j \rightarrow N^p N^q) = \frac{1}{\Pr(w_{1,n})} \sum_{k,l,m} \alpha_j(k, l) \cdot \Pr(N^j \rightarrow N^p N^q) \cdot \beta_p(k, m) \cdot \beta_q(m+1, l)$$

$$C(N^i \rightarrow w^j) = \frac{1}{\Pr(w_{1,n})} \sum_k \alpha_i(k, k) \cdot \Pr(N^i \rightarrow w^j, w^j = w^k)$$



## Parsing

- Parsing mit modifiziertem EARLEY/CYK-Algorithmus
- dynamische Programmierung:
  - rekursiver Aufbau der Parsingtabelle und Auswahl der lokal optimalen Interpretation



## Erweitertes Basismodell

- Problem: Unabhängigkeitsannahme ist systematisch falsch
  - Subjekt wird häufiger pronominalisiert als Objekt
    - besonders deutlich in gesprochener Sprache
    - Auswirkung der Informationsstruktur
  - Subkategorisierungspräferenzen disambiguieren Anbindungsprobleme
    - NP Anbindung ist häufiger als V-Anbindung (2:1)
    - aber: einige Verben erzwingen Anbindung bestimmter Präpositionen
- Moscow sent more than 100.000 soldiers into Afghanistan.*
  - send requires a direction (into)
    - Modellierung lexikalischer Abhängigkeiten



## Erweitertes Basismodell

- lexikalische Abhängigkeiten lassen sich in PCFG nicht ausdrücken
  - nur stochastische Abhängigkeit vom dominierenden Nichtterminal
 
$$\Pr(N \rightarrow \zeta | N)$$
- Erweiterung des stochastischen Modells um zusätzliche Bedingungen



## Erweitertes Basismodell

- lexikalisierte Regelanwendungswahrscheinlichkeiten (CHARNIAK 2000)

$$\Pr(N \rightarrow \zeta | N, h(r))$$

- lexikalische Abhängigkeiten (CHARNIAK 2000, COLLINS 1999)

- vom Kopf der unmittelbar übergeordneten Phrase

$$\Pr(r = N \rightarrow \zeta | N, h(r), h(m(r)))$$

- vom Kopf der beiden übergeordneten Phrasenebenen

$$\Pr(r = N \rightarrow \zeta | N, h(r), h(m(r)), h(m(m(r))))$$

Wolfgang Menzel: Sprachorientierte KI: Syntax und Parsing – p. 87

## Erweitertes Basismodell

- Problem: Datenmangel
  - Backoff
  - Glätten
  - stochastische Modellierung der Schwesterknoten zum Kopf als Markov-Prozess (COLLINS 1999)

Wolfgang Menzel: Sprachorientierte KI: Syntax und Parsing – p. 87

## Erweitertes Basismodell

- Qualität (CHARNIAK 2000)

Satzlänge ≤ 40					
Parser	LR	LP	CB	0CB	2CB
COLLINS 1999	88.5	88.7	0.92	66.7	87.1
CHARNIAK 2000	90.1	90.1	0.74	70.1	89.6
Satzlänge ≤ 100					
Parser	LR	LP	CB	0CB	2CB
COLLINS 1999	88.1	88.3	1.06	64.0	85.1
CHARNIAK 2000	89.6	89.5	0.88	67.6	87.7

Wolfgang Menzel: Sprachorientierte KI: Syntax und Parsing – p. 88

## Alternative Modelle

- Parsing als stochastischer Mustervergleich (MAGERMAN 1994)
  - "Parsing ohne Grammatik"
- Training eines binären Entscheidungsbaumes (decision tree)
  - strukturelle Anbindung von Wortformen und Teilbäumen
  - Wahl der strukturellen Kategorie

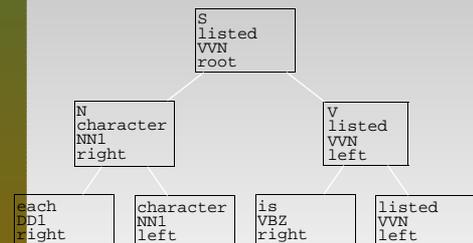
Wolfgang Menzel: Sprachorientierte KI: Syntax und Parsing – p. 89

## Alternative Modelle

- Merkmalsrepräsentation für Syntaxbäume
  - label: phrasale Kategorie (N,V,S, ...) nur für nichtterminale Knoten
  - word: Wortform des Blattknotens bzw. des Konstituentenkopfes
  - tag: Wortart des Blattknotens bzw. des Konstituentenkopfes
  - extension: linke oder rechte Konstituentengrenze, bzw. mittlere Position (right, left, up)

Wolfgang Menzel: Sprachorientierte KI: Syntax und Parsing – p. 90

## Alternative stochastische Modelle



Wolfgang Menzel: Sprachorientierte KI: Syntax und Parsing – p. 91

## Alternative Modelle

- Training des Entscheidungsbaumes
  - Beschränkung der initialen Knotenmenge:
    - bottom-up/links-rechts Derivation: Ein Knoten wird erst konstruiert, wenn unterhalb und links davon bereits alle Informationen verfügbar sind
    - Ableitungsfenster: es werden immer nur  $n$  mögliche Merkmalswertzuweisungen betrachtet (→ aktive Knoten) tatsächliche Fenstergröße: 1 ... 2
  - Ziel: Optimale Reihenfolge von Entscheidungsfragen
  - Optimierungskriterium: Maximale Entropiereduktion für die Trainingsmenge

Wolfgang Menzel: Sprachorientierte KI: Syntax und Parsing – p. 92

## Alternative Modelle

- Resultate
  - Trainingsdaten: Baumkorpus mit 1473 Sätzen
  - 78% korrekte Klammerstruktur (crossing-brackets score)
  - 35% korrekte Baumstruktur (mit Wortartentagging)
  - 50% korrekte Baumstruktur (ohne Wortartentagging)
  - Vergleich: 69% korrekte Klammerstruktur für klassischen Parser

Wolfgang Menzel: Sprachorientierte KI: Syntax und Parsing – p. 93

## Alternative Modelle

- Datenorientiertes Parsing (DOP) (BOD 1992, 2003)
  - Zerlegung der Parsebäume in Teilbäume bis zu einer maximalen Höhe  $n$  ( $n \leq 6$ )
  - Schätzen der Auftretenswahrscheinlichkeit aller Teilbäume
  - Ermitteln der Ableitungswahrscheinlichkeit für eine Ergebnisstruktur als Summe über alle Ableitungsvarianten
  - geschlossene Berechnung nicht mehr möglich  
→ Monte-Carlo-sampling
  - LR=90.7%, LP=90.8% (Satzlänge  $\leq 100$ )



## Alternative Modelle

- Supertagging (BANGALORE 1997)
  - Zerlegen des Parsebaumes in lexikalisierte Baumfragmente
    - analog zu einer Tree Adjoining Grammar (TAG)
  - Verwendung der Baumfragmente als strukturell reichhaltige lexikalische Kategorien
  - Training eines stochastischen Taggers
  - Auswahl der wahrscheinlichsten Folge von Strukturfragmenten  
→ almost parsing



## Alternative Modelle

- Supertagging
  - Rekonstruktion eines Parsebaumes aus den Baumfragmenten
  - bessere Resultate (geringere Perplexität) mit einer Constraint Dependency Grammar (HARPER 2002)
    - auch bei Training auf fehlerhaften Baumbanken (HARPER 2003)



## Stochastisches Parsing

- Anwendungen:
  - approximatives Parsing für unrestringierten Text
    - Informationsextraktion
    - Diskursanalyse
  - Analyse ungrammatischer Äußerungen
  - Sprachmodelle in der Spracherkennung
  - Lernen von Grammatiken

