

Grundlegende Parsingalgorithmen

Chart-Parsing: Left-Corner-Parser

Kurt Eberle

k.eberle@lingenio.de

(Viele Folien, Teile von Folien, Materialien von **Helmut Schmid's**
Parsing-Kurs WS14 Tübingen, u.a.)

27. Februar 2020

Überblick

Left-Corner-Parser

Left-Corner-Erkennen I

Aktionen des Left-Corner-Erkenners

Top-Down-Filter

Left-Corner-Erkennen II

Aktionen des Left-Corner-Erkenners II

Überblick

Left-Corner-Parser

Left-Corner-Erkennen I

Aktionen des Left-Corner-Erkenners

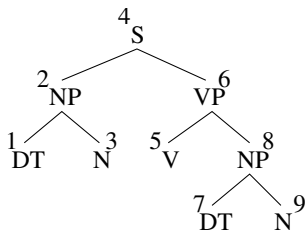
Top-Down-Filter

Left-Corner-Erkennen II

Aktionen des Left-Corner-Erkenners II

Left-Corner-Parser

- ▶ Left-Corner-Parser:



- ▶ Left-Corner-Parser: Parser konstruiert Parses in Left-Corner-Ordnung.
- ▶ Der Left-Corner-Parser ist ein Bottom-up-Chart-Parser.
- ▶ Hauptunterschied zum Earley-Parser: Die Predikt-Operation wird über passive Punktregeln statt über aktive getriggert.

Überblick

Left-Corner-Parser

Left-Corner-Erkennen I

Aktionen des Left-Corner-Erkenners

Top-Down-Filter

Left-Corner-Erkennen II

Aktionen des Left-Corner-Erkenners II

Left-Corner-Erkennen I

- ▶ Grammatik ohne ε -Produktionen
- ▶ Terminale Symbole werden über unäre Regeln generiert ($A \rightarrow a$).

Recognizer($x_1 \dots x_n$)

$t_0 := \{\}$

For $j := 1$ to n

$t_j := \{\}$

For all $A \rightarrow x_j \in P$

Store($A \rightarrow x_j \cdot, j - 1, j$)

If $\langle S \rightarrow \alpha \cdot, 0, n \rangle \in t_n$ for some α then

accept

else

report error

Überblick

Left-Corner-Parser

Left-Corner-Erkennen I

Aktionen des Left-Corner-Erkenners

Top-Down-Filter

Left-Corner-Erkennen II

Aktionen des Left-Corner-Erkenners II

Aktionen des Left-Corner-Erkenners

Store($A \rightarrow \alpha \cdot \beta, i, k$)

If $\langle A \rightarrow \alpha \cdot \beta, i, k \rangle \notin t_k$ then
 $t_k := t_k \cup \{ \langle A \rightarrow \alpha \cdot \beta, i, k \rangle \}$
 If $\beta = \varepsilon$ then
 Complete(A, i, k)
 Predict(A, i, k)

Complete(A, j, k)

If called2[A, j, k] = false then
 called2[A, j, k] := true
 For all $\langle B \rightarrow \beta \cdot A\gamma, i, j \rangle \in t_j$
 Store($B \rightarrow \beta A \cdot \gamma, i, k$)

Aktionen des Left-Corner-Erkenners

Predict(A, i, k)

If $\text{called1}[A, i, k] = \text{false}$ then

$\text{called1}[A, i, k] := \text{true}$

 For all $B \rightarrow A\alpha \in P$

 Store($B \rightarrow A \cdot \alpha, i, k$)

Überblick

Left-Corner-Parser

Left-Corner-Erkennen I

Aktionen des Left-Corner-Erkenners

Top-Down-Filter

Left-Corner-Erkennen II

Aktionen des Left-Corner-Erkenners II

Top-Down-Filter

- ▶ Problem: Die bisher beschriebene Form des Left-Corner-Parsers führt viele unnötige Punktregeln ein.
- ▶ Lösung: Top-down-Filter
- ▶ Erreichbarkeitsfunktion R
$$R(A) := \{B \mid A \xRightarrow{*} B\alpha\}$$
- ▶ Der Prediktor führt eine neue Punktregel $\langle B \rightarrow A \cdot \alpha, j, k \rangle$ nur dann ein, wenn eine Punktregel $\langle C \rightarrow \beta \cdot D\gamma, i, j \rangle$ existiert so, dass $B \in R(D)$ ist.

Überblick

Left-Corner-Parser

Left-Corner-Erkennen I

Aktionen des Left-Corner-Erkenners

Top-Down-Filter

Left-Corner-Erkennen II

Aktionen des Left-Corner-Erkenners II

Left-Corner-Erkenner II

Recognizer($x_1 \dots x_n$)

> $r_0 := R(S)$

For $j := 1$ to n

$t_j := \{\}$

> $r_j := \{\}$

> For all $A \rightarrow x_j \in P$ such that $A \in r_{j-1}$

Store($A \rightarrow x_j \cdot, j-1, j$)

If $\langle S \rightarrow \alpha \cdot, 0, n \rangle \in t_n$ for some α then

accept

else

report error

Überblick

Left-Corner-Parser

Left-Corner-Erkennen I

Aktionen des Left-Corner-Erkenners

Top-Down-Filter

Left-Corner-Erkennen II

Aktionen des Left-Corner-Erkenners II

Aktionen des Left-Corner-Erkenners II

Store($A \rightarrow \alpha \cdot \beta, i, k$)

If $\langle A \rightarrow \alpha \cdot \beta, i, k \rangle \notin t_k$ then

$t_k := t_k \cup \{\langle A \rightarrow \alpha \cdot \beta, i, k \rangle\}$

If $\beta = \varepsilon$ then

Complete(A, i, k)

Predict(A, i, k)

> else if $\beta = B\gamma$ for some B, γ then

> $r_k := r_k \cup R(B)$

Aktionen des Left-Corner-Erkenners II

Complete(A, j, k)

If $\text{called2}[A, j, k] = \text{false}$ then

$\text{called2}[A, j, k] := \text{true}$

For all $\langle B \rightarrow \beta \cdot A \gamma, i, j \rangle \in t_j$

$\text{Store}(B \rightarrow \beta A \cdot \gamma, i, k)$

Predict(A, i, k)

If $\text{called1}[A, i, k] = \text{false}$ then

$\text{called1}[A, i, k] := \text{true}$

> For all $B \rightarrow A \alpha \in P$ such that $B \in r_i$

$\text{Store}(B \rightarrow A \cdot \alpha, i, k)$