Formal Theory of Context-Free Grammars

Initial Bachelor Seminar Talk

Jana Hofmann

Advisor: Prof. Dr. Gert Smolka

SAARLAND UNIVERSITY

COMPUTER SCIENCE

29th May 2015

Topics

Formalization of Context-Free Grammars in Coq

Verified Algorithm for Normalization

(Decidability of Context-Free Languages)

Sources

Dexter C. Kozen
Automata and Computability
Springer, 1997

John E. Hopcroft, Rajeev Motwani and Jeffrey D. Ullman
Introduction to automata theory, languages, and computation
AddisonWesley, 2nd edition, 2001

Denis Firsov and Tarmo Uustalu
Certified Normalization of Context-Free Grammars
Institute of Cybernetics at TUT, 2015

Jan-Oliver Kaiser
Constructive Formalization of Regular Languages
Bachelor thesis, Saarland University, 2012

Content

Context-Free Grammars

Context-Free Grammars are used to

- describe non-regular languages
- define programming languages (Backus-Naur-Form)

Example

$A \longrightarrow \varepsilon$
$A \longrightarrow (\ A\ )$
$A \longrightarrow AA$

describes $\{\varepsilon, (), (()), ()(), (())(), \ldots\}$

Notation

	notation	example
variable	$A, B, C, \ldots$	A
terminal	$a, b, c, \ldots$	(,)
phrase	u, v, w	$(\ A\ )$, $(\ A\ (\ AA\ )\ )$, ε
rule	r	$A \longrightarrow (\ A\ )$
grammar	G	$A \longrightarrow \varepsilon \mid (\ A\ ) \mid AA$
grammar with start symbol	(G, S)	

Chomsky Normal Form

- Chomsky Normal Form is the foundation for further reasoning on CFGs (e.g. CYK algorithm)

(G, S) is in Chomsky Normal form if every rule in G is of one of the following forms:

- $A \longrightarrow BC$ where $B, C \neq S$
- $A \longrightarrow a$
- $S \longrightarrow \varepsilon$

Chomsky Normal Form

Example

$$A \longrightarrow \varepsilon \mid (\ A\ ) \mid AA \quad \rightsquigarrow \quad \begin{aligned} A &\longrightarrow \varepsilon \mid B\ ) \mid AA \\ B &\longrightarrow\ (\ A \end{aligned}$$

$$\rightsquigarrow \quad \begin{aligned} A &\longrightarrow \varepsilon \mid BC \mid AA \\ B &\longrightarrow DA \\ C &\longrightarrow\ ) \\ D &\longrightarrow\ (\end{aligned}$$

Definitions

$$\begin{aligned} \mathit{var} &:= n & (n \in \mathbb{N}) \\ \mathit{ter} &:= n & (n \in \mathbb{N}) \\ \mathit{symbol} &:= \mathit{var} \mid \mathit{ter} \end{aligned}$$

$$\begin{aligned} \mathit{phrase} &:= \mathscr{L}(\mathit{symbol}) \\ \mathit{rule} &:= \mathit{var} \times \mathit{phrase} \\ \mathit{grammar} &:= \mathscr{L}(\mathit{rule}) \end{aligned}$$

Derivation

- $\Longrightarrow$: $grammar \to var \to phrase \to Prop$

$$\frac{}{A \overset{G}{\Longrightarrow} A} \qquad \frac{A \longrightarrow u \in G}{A \overset{G}{\Longrightarrow} u} \qquad \frac{A \overset{G}{\Longrightarrow} uBw \quad B \overset{G}{\Longrightarrow} v}{A \overset{G}{\Longrightarrow} uvw}$$

- $\mathcal{L}$: $grammar \to var \to phrase \to Prop$

$$\mathcal{L}_G^A := \lambda u.\ (A \overset{G}{\Longrightarrow} u \ \wedge \ terminal\ \ u)$$

Transformation into CNF

1. eliminate all ε-rules ($A \longrightarrow \varepsilon$)
2. eliminate unit-rules ($A \longrightarrow B$)
3. eliminate long-rules ($A \longrightarrow X_1 X_2 ... X_k$)
4. replace terminals with variables

ε- Elimination

1. add new rules by dropping variables

Example

$$A \longrightarrow \varepsilon \mid a \qquad\qquad A \longrightarrow \varepsilon \mid a$$

$$B \longrightarrow \varepsilon \mid b \qquad \rightsquigarrow \qquad B \longrightarrow \varepsilon \mid b$$

$$C \longrightarrow AB \mid cAc \qquad\qquad C \longrightarrow AB \mid \varepsilon \mid A \mid B \mid cAc \mid cc$$

2. remove all rules of the form $A \longrightarrow \varepsilon$ (and add $S \longrightarrow \varepsilon$)

1) Adding Nullable Rules

- construct the closure G' of G with respect to

$$\frac{A \longrightarrow u \in G}{A \longrightarrow u \in G'} \qquad \frac{A \longrightarrow u_1 X u_2 \in G' \quad X \longrightarrow \varepsilon \in G'}{A \longrightarrow u_1 u_2 \in G'}$$

- prove that $\mathcal{L}_G^A \equiv \mathcal{L}_{closure\ \ G}^A$

ε- Elimination

$$\frac{A \longrightarrow u \in G}{A \longrightarrow u \in G'} \qquad \frac{A \longrightarrow u_1 X u_2 \in G' \quad X \longrightarrow \varepsilon \in G'}{A \longrightarrow u_1 u_2 \in G'}$$

- fixpoint iteration
- termination: new rule is a subset of the old so only finitely many rules can be added
- in Coq bounded recursion with two bounds:
  - number of possible rules
  - $|G'|$ - (number of steps done without adding a rule)

Correctness

$A \overset{G}{\Longrightarrow} u \ \leftrightarrow \ A \overset{G'}{\Longrightarrow} u$

$\rightarrow$: easy

$\leftarrow$: essential:

let $r \in \mathit{closure}\ G$, $r \notin G$

prove: $A \overset{r::G}{\Longrightarrow} u \ \rightarrow \ A \overset{G}{\Longrightarrow} u$

induction on $A \overset{r::G}{\Longrightarrow} u$:

- $u = A$ and we get $A \overset{G}{\Longrightarrow} A$
- $A \longrightarrow u \in r :: G$. So either $A \longrightarrow u \in G$ (trivial) or $r = A \longrightarrow u$
  then by construction
  $\exists\ u_1\ u_2\ X.\ A \longrightarrow u_1 X u_2 \in G \ \wedge \ X \longrightarrow \varepsilon \in G \ \wedge u = u_1 u_2$.
  So we get $A \overset{G}{\Longrightarrow} u$.
- $u = u_1 u_2 u_3$. By IH: $A \overset{G}{\Longrightarrow} u_1 X u_3$, $X \overset{G}{\Longrightarrow} u_2$ so $A \overset{G}{\Longrightarrow} u$.

Correctness (2)

1)

$$nullable_G^A \ := \ A \xRightarrow{G} \varepsilon$$

2)

$$nullable'^A_G \ := \ A \longrightarrow \varepsilon \in closure \ G$$

3)

$$\frac{\forall\, X \in u.\ nullable''^X_G \quad A \longrightarrow u \in G}{nullable''^A_G}$$

- 1) $\leftrightarrow$ 2) proof not yet done
- 1) $\leftrightarrow$ 3) proof by mutual induction

2) Deleting ε - Rules

- every rule $B \longrightarrow \varepsilon$ is superfluous now
- $\mathcal{L}_G^A - \{\varepsilon\} \equiv \mathcal{L}_{G' - \{B \longrightarrow \varepsilon\}}^A$
- Proof: Let $A \overset{G}{\Longrightarrow} u$ be of minimal length and $u \neq \varepsilon$.
  no rule $B \longrightarrow \varepsilon$ needed (otherwise not minimal length)
- if $\varepsilon \in \mathcal{L}_G^S$, add $S \longrightarrow \varepsilon$

$\Rightarrow \mathcal{L}_G^S \equiv \mathcal{L}_{closure\ G}^S$

Outlook

- proof for *nullable* correctness property
- proof for deletion of ε - Rules
- finish algorithm and it's verification:
  - deletion of unit-rules
  - deletion of long-rules
  - new rules for terminals
- add other constraints to CNF (e.g. useless symbols)
- decidability of context-free languages: CYK-algorithm

Derivations

$$\frac{}{A \overset{G}{\Longrightarrow} A} \qquad \frac{A \longrightarrow u \in G}{A \overset{G}{\Longrightarrow} u} \qquad \frac{A \overset{G}{\Longrightarrow} uBw \quad B \overset{G}{\Longrightarrow} v}{A \overset{G}{\Longrightarrow} uvw}$$

is equivalent to

$$\frac{}{A \overset{G}{\Longrightarrow} [A]} \qquad \frac{A \overset{G}{\Longrightarrow} uBw \quad B \longrightarrow v \in G}{A \overset{G}{\Longrightarrow} uvw}$$

proof by straightforward induction

Equivalence nullable

strengthen the statement:
$A \overset{G}{\Longrightarrow} u \rightarrow \forall\ X \in u.\ X \overset{G}{\Longrightarrow} \varepsilon \rightarrow \mathit{nullable}''^{G}_{A}$

proof by induction on $A \overset{G}{\Longrightarrow} u$.