CS 383

Lecture 10 – Chomsky Normal Form

Stephen Checkoway

Fall, 2023

More CFLs

- $A = \{\mathtt{a}^i\mathtt{b}^j\mathtt{c}^k \mid i \le j \text{ or } i = k\}$
- $B = \{w \mid w \in \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}^*$ contains the same number of as as bs and cs combined$\}$
- $C = \{\mathtt{1}^m\mathtt{+}\mathtt{1}^n\mathtt{=}\mathtt{1}^{m+n} \mid m, n \ge 1\}$; $\Sigma = \{\mathtt{1}, \mathtt{+}, \mathtt{=}\}$
- $D = \overline{(\mathtt{abb}^* \mid \mathtt{bbaa})^*}$
- $E = \{w \mid w \in \{\mathtt{0}, \mathtt{1}\}^*$ and $w^{\mathcal{R}}$ is a binary number not divisible by 5$\}$

Another proof that regular languages are context-free

We can encode the computation of a DFA on a string using a CFG

Give a DFA $M = (Q, \Sigma, \delta, q_0, F)$, we can construct an equivalent CFG $G = (V, \Sigma, R, S)$ where

Another proof that regular languages are context-free

We can encode the computation of a DFA on a string using a CFG

Give a DFA $M = (Q, \Sigma, \delta, q_0, F)$, we can construct an equivalent CFG $G = (V, \Sigma, R, S)$ where

- states of M are variables in G

Another proof that regular languages are context-free

We can encode the computation of a DFA on a string using a CFG

Give a DFA $M = (Q, \Sigma, \delta, q_0, F)$, we can construct an equivalent CFG $G = (V, \Sigma, R, S)$ where

- states of M are variables in G
- q_0 is the start variable, and

Another proof that regular languages are context-free

We can encode the computation of a DFA on a string using a CFG

Give a DFA $M = (Q, \Sigma, \delta, q_0, F)$, we can construct an equivalent CFG $G = (V, \Sigma, R, S)$ where

- states of M are variables in G
- q_0 is the start variable, and
- transitions $\delta(q, t) = r$ become rules $q \to tr$

Another proof that regular languages are context-free

We can encode the computation of a DFA on a string using a CFG

Give a DFA $M = (Q, \Sigma, \delta, q_0, F)$, we can construct an equivalent CFG $G = (V, \Sigma, R, S)$ where

- states of M are variables in G
- q_0 is the start variable, and
- transitions $\delta(q, t) = r$ become rules $q \to tr$

If on input $w = w_1 w_2 \cdots w_n$, M goes through states $r_0, r_1, \ldots, r_n$, then

$$r_0 \Rightarrow w_1 r_1 \Rightarrow w_1 w_2 r_2 \Rightarrow \cdots \Rightarrow w_1 w_2 \cdots w_n r_n$$

Another proof that regular languages are context-free

We can encode the computation of a DFA on a string using a CFG

Give a DFA $M = (Q, \Sigma, \delta, q_0, F)$, we can construct an equivalent CFG $G = (V, \Sigma, R, S)$ where

- states of M are variables in G
- q_0 is the start variable, and
- transitions $\delta(q, t) = r$ become rules $q \to tr$

If on input $w = w_1 w_2 \cdots w_n$, M goes through states $r_0, r_1, \ldots, r_n$, then

$$r_0 \Rightarrow w_1 r_1 \Rightarrow w_1 w_2 r_2 \Rightarrow \cdots \Rightarrow w_1 w_2 \cdots w_n r_n$$

So G has derived the string $w r_n$ but this still has a variable

What additional rules should we add to end up with a string of terminals?

For each state $q \in F$, add a rule $q \to \varepsilon$

Formally

Proof.

Given a DFA $M = (Q, \Sigma, \delta, q_0, F)$, we can construct an equivalent CFG $G = (V, \Sigma, R, S)$ where

$$V = Q$$

$$S = q_0$$

$$R = \{q \to tr \ : \ \delta(q, t) = r\} \cup \{q \to \varepsilon \ : \ q \in F\}$$

Formally

Proof.

Given a DFA $M = (Q, \Sigma, \delta, q_0, F)$, we can construct an equivalent CFG $G = (V, \Sigma, R, S)$ where

$$
\begin{aligned}
V &= Q \\
S &= q_0 \\
R &= \{q \to tr \;:\; \delta(q,t) = r\} \cup \{q \to \varepsilon \;:\; q \in F\}
\end{aligned}
$$

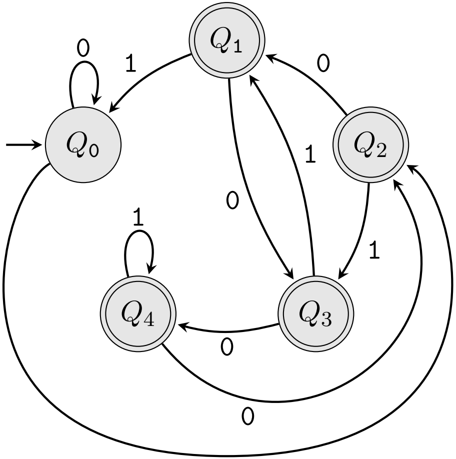
If $r_0, r_1, \ldots, r_n$ is the computation of M on input $w = w_1 w_2 \cdots w_n$, then $r_0 = q_0$ and $\delta(r_{i-1}, w_i) = r_i$ for $1 \le i \le n$

By construction $r_0 \Rightarrow w_1 r_1 \Rightarrow w_1 w_2 r_2 \overset{*}{\Rightarrow} w_1 w_2 \cdots w_n r_n$

Therefore, $w \in L(M)$ iff $r_n \in F$ iff $r_n \Rightarrow \varepsilon$ iff $q_0 \overset{*}{\Rightarrow} w$ iff $w \in L(G)$ $\square$

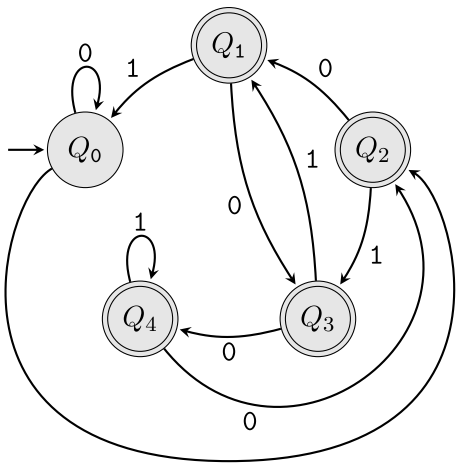
Returning to our language

$$E = \{w \mid w \in \{\mathtt{0},\mathtt{1}\}^* \text{ and } w^{\mathcal{R}} \text{ is a binary number not divisible by 5}\}$$



Returning to our language

$E = \{w \mid w \in \{\mathtt{0},\mathtt{1}\}^* \text{ and } w^{\mathcal{R}} \text{ is a binary number not divisible by 5}\}$



$$
\begin{aligned}
Q_0 &\to \mathtt{0}Q_0 \mid \mathtt{1}Q_2 \\
Q_1 &\to \mathtt{0}Q_3 \mid \mathtt{1}Q_0 \mid \varepsilon \\
Q_2 &\to \mathtt{0}Q_1 \mid \mathtt{1}Q_3 \mid \varepsilon \\
Q_3 &\to \mathtt{0}Q_4 \mid \mathtt{1}Q_1 \mid \varepsilon \\
Q_4 &\to \mathtt{0}Q_2 \mid \mathtt{1}Q_4 \mid \varepsilon
\end{aligned}
$$

Chomsky Normal Form (CNF)

A CFG $G = (V, \Sigma, R, S)$ is in Chomsky Normal Form if all rules have one of these forms

- $S \to \varepsilon$ where S is the start variable
- $A \to BC$ where $A \in V$ and $B, C \in V \setminus \{S\}$
- $A \to t$ where $A \in V$ and $t \in \Sigma$

Note

- The only rule with ε on the right has the start variable on the left
- The start variable doesn't appear on the right hand side of any rule

CNF example

Let $A = \{w \mid w \in \{\texttt{a}, \texttt{b}\}^* \text{ and } w = w^{\mathcal{R}}\}$.

CFG in CNF

$$
\begin{aligned}
S &\to AU \mid BV \mid \texttt{a} \mid \texttt{b} \mid \varepsilon \\
T &\to AU \mid BV \mid \texttt{a} \mid \texttt{b} \\
U &\to TA \mid \texttt{a} \\
V &\to TB \mid \texttt{b} \\
A &\to \texttt{a} \\
B &\to \texttt{b}
\end{aligned}
$$

Derivation of baaab

$$S$$

CNF example

Let $A = \{w \mid w \in \{\mathtt{a}, \mathtt{b}\}^* \text{ and } w = w^{\mathcal{R}}\}$.

CFG in CNF

$$
\begin{aligned}
S &\to AU \mid BV \mid \mathtt{a} \mid \mathtt{b} \mid \varepsilon \\
T &\to AU \mid BV \mid \mathtt{a} \mid \mathtt{b} \\
U &\to TA \mid \mathtt{a} \\
V &\to TB \mid \mathtt{b} \\
A &\to \mathtt{a} \\
B &\to \mathtt{b}
\end{aligned}
$$

Derivation of baaab

$$S \Rightarrow BV$$

CNF example

Let $A = \{w \mid w \in \{\mathtt{a}, \mathtt{b}\}^* \text{ and } w = w^{\mathcal{R}}\}$.

CFG in CNF

$$
\begin{aligned}
S &\to AU \mid BV \mid \mathtt{a} \mid \mathtt{b} \mid \varepsilon \\
T &\to AU \mid BV \mid \mathtt{a} \mid \mathtt{b} \\
U &\to TA \mid \mathtt{a} \\
V &\to TB \mid \mathtt{b} \\
A &\to \mathtt{a} \\
B &\to \mathtt{b}
\end{aligned}
$$

Derivation of baaab

$$
\begin{aligned}
S &\Rightarrow BV \\
&\Rightarrow \mathtt{b}V
\end{aligned}
$$

CNF example

Let $A = \{w \mid w \in \{\mathtt{a}, \mathtt{b}\}^* \text{ and } w = w^{\mathcal{R}}\}$.

CFG in CNF

$$
\begin{aligned}
S &\to AU \mid BV \mid \mathtt{a} \mid \mathtt{b} \mid \varepsilon \\
T &\to AU \mid BV \mid \mathtt{a} \mid \mathtt{b} \\
U &\to TA \mid \mathtt{a} \\
V &\to TB \mid \mathtt{b} \\
A &\to \mathtt{a} \\
B &\to \mathtt{b}
\end{aligned}
$$

Derivation of baaab

$$
\begin{aligned}
S &\Rightarrow BV \\
&\Rightarrow \mathtt{b}V \\
&\Rightarrow \mathtt{b}TB
\end{aligned}
$$

CNF example

Let $A = \{w \mid w \in \{\texttt{a}, \texttt{b}\}^* \text{ and } w = w^{\mathcal{R}}\}$.

CFG in CNF

$$
\begin{aligned}
S &\to AU \mid BV \mid \texttt{a} \mid \texttt{b} \mid \varepsilon \\
T &\to AU \mid BV \mid \texttt{a} \mid \texttt{b} \\
U &\to TA \mid \texttt{a} \\
V &\to TB \mid \texttt{b} \\
A &\to \texttt{a} \\
B &\to \texttt{b}
\end{aligned}
$$

Derivation of baaab

$$
\begin{aligned}
S &\Rightarrow BV \\
&\Rightarrow \texttt{b}V \\
&\Rightarrow \texttt{b}TB \\
&\Rightarrow \texttt{b}AUB
\end{aligned}
$$

CNF example

Let $A = \{w \mid w \in \{\mathtt{a}, \mathtt{b}\}^* \text{ and } w = w^{\mathcal{R}}\}$.

CFG in CNF

$$
\begin{aligned}
S &\to AU \mid BV \mid \mathtt{a} \mid \mathtt{b} \mid \varepsilon \\
T &\to AU \mid BV \mid \mathtt{a} \mid \mathtt{b} \\
U &\to TA \mid \mathtt{a} \\
V &\to TB \mid \mathtt{b} \\
A &\to \mathtt{a} \\
B &\to \mathtt{b}
\end{aligned}
$$

Derivation of baaab

$$
\begin{aligned}
S &\Rightarrow BV \\
&\Rightarrow \mathtt{b}V \\
&\Rightarrow \mathtt{b}TB \\
&\Rightarrow \mathtt{b}AUB \\
&\Rightarrow \mathtt{ba}UB
\end{aligned}
$$

CNF example

Let $A = \{w \mid w \in \{\mathtt{a}, \mathtt{b}\}^* \text{ and } w = w^{\mathcal{R}}\}$.

CFG in CNF

$$
\begin{aligned}
S &\to AU \mid BV \mid \mathtt{a} \mid \mathtt{b} \mid \varepsilon \\
T &\to AU \mid BV \mid \mathtt{a} \mid \mathtt{b} \\
U &\to TA \mid \mathtt{a} \\
V &\to TB \mid \mathtt{b} \\
A &\to \mathtt{a} \\
B &\to \mathtt{b}
\end{aligned}
$$

Derivation of baaab

$$
\begin{aligned}
S &\Rightarrow BV \\
&\Rightarrow \mathtt{b}V \\
&\Rightarrow \mathtt{b}TB \\
&\Rightarrow \mathtt{b}AUB \\
&\Rightarrow \mathtt{ba}UB \\
&\Rightarrow \mathtt{ba}TAB
\end{aligned}
$$

CNF example

Let $A = \{w \mid w \in \{\mathtt{a}, \mathtt{b}\}^* \text{ and } w = w^{\mathcal{R}}\}$.

CFG in CNF

$$
\begin{aligned}
S &\to AU \mid BV \mid \mathtt{a} \mid \mathtt{b} \mid \varepsilon \\
T &\to AU \mid BV \mid \mathtt{a} \mid \mathtt{b} \\
U &\to TA \mid \mathtt{a} \\
V &\to TB \mid \mathtt{b} \\
A &\to \mathtt{a} \\
B &\to \mathtt{b}
\end{aligned}
$$

Derivation of baaab

$$
\begin{aligned}
S &\Rightarrow BV \\
&\Rightarrow \mathtt{b}V \\
&\Rightarrow \mathtt{b}TB \\
&\Rightarrow \mathtt{b}AUB \\
&\Rightarrow \mathtt{ba}UB \\
&\Rightarrow \mathtt{ba}TAB \\
&\Rightarrow \mathtt{baa}AB
\end{aligned}
$$

CNF example

Let $A = \{w \mid w \in \{\mathtt{a}, \mathtt{b}\}^* \text{ and } w = w^{\mathcal{R}}\}$.

CFG in CNF

$$
\begin{aligned}
S &\to AU \mid BV \mid \mathtt{a} \mid \mathtt{b} \mid \varepsilon \\
T &\to AU \mid BV \mid \mathtt{a} \mid \mathtt{b} \\
U &\to TA \mid \mathtt{a} \\
V &\to TB \mid \mathtt{b} \\
A &\to \mathtt{a} \\
B &\to \mathtt{b}
\end{aligned}
$$

Derivation of baaab

$$
\begin{aligned}
S &\Rightarrow BV \\
&\Rightarrow \mathtt{b}V \\
&\Rightarrow \mathtt{b}TB \\
&\Rightarrow \mathtt{b}AUB \\
&\Rightarrow \mathtt{ba}UB \\
&\Rightarrow \mathtt{ba}TAB \\
&\Rightarrow \mathtt{baa}AB \\
&\Rightarrow \mathtt{baaa}B
\end{aligned}
$$

CNF example

Let $A = \{w \mid w \in \{\mathtt{a}, \mathtt{b}\}^* \text{ and } w = w^{\mathcal{R}}\}$.

CFG in CNF

$$
\begin{aligned}
S &\to AU \mid BV \mid \mathtt{a} \mid \mathtt{b} \mid \varepsilon \\
T &\to AU \mid BV \mid \mathtt{a} \mid \mathtt{b} \\
U &\to TA \mid \mathtt{a} \\
V &\to TB \mid \mathtt{b} \\
A &\to \mathtt{a} \\
B &\to \mathtt{b}
\end{aligned}
$$

Derivation of $\mathtt{baaab}$

$$
\begin{aligned}
S &\Rightarrow BV \\
&\Rightarrow \mathtt{b}V \\
&\Rightarrow \mathtt{b}TB \\
&\Rightarrow \mathtt{b}AUB \\
&\Rightarrow \mathtt{ba}UB \\
&\Rightarrow \mathtt{ba}TAB \\
&\Rightarrow \mathtt{baa}AB \\
&\Rightarrow \mathtt{baaa}B \\
&\Rightarrow \mathtt{baaab}
\end{aligned}
$$

Converting to CNF

Theorem

Every context-free language A is generated by some CFG in CNF.

Proof.

Given a CFG $G = (V, \Sigma, R, S)$ generating A, we construct a new CFG $G' = (V', \Sigma, R', S')$ in CNF generating A.

There are five steps.

- START Add a new start variable
- BIN Replace rules with RHS longer than two with multiple rules each of which has a RHS of length two
- DEL-ε Remove all ε-rules $(A \to \varepsilon)$
- UNIT Remove all unit-rules $(A \to B)$
- TERM Add a variable and rule for each terminal $(T \to t)$ and replace terminals on the RHS of rules

Proof continued

In the following $x \in V \cup \Sigma$ and $u \in (\Sigma \cup V)^+$

START Add a new start variable S' and a rule $S' \rightarrow S$

Proof continued

In the following $x \in V \cup \Sigma$ and $u \in (\Sigma \cup V)^{+}$

START Add a new start variable S' and a rule $S' \rightarrow S$

BIN Replace each rule $A \rightarrow xu$ with the rules $A \rightarrow xA_1$ and $A_1 \rightarrow u$ and repeat until the RHS of every rule has length at most two

Proof continued

In the following $x \in V \cup \Sigma$ and $u \in (\Sigma \cup V)^+$

- START Add a new start variable S' and a rule $S' \to S$
- BIN Replace each rule $A \to xu$ with the rules $A \to xA_1$ and $A_1 \to u$ and repeat until the RHS of every rule has length at most two
- DEL-ε For each rule of the form $A \to \varepsilon$ other than $S' \to \varepsilon$ remove $A \to \varepsilon$ and update all rules with A in the RHS

Proof continued

In the following $x \in V \cup \Sigma$ and $u \in (\Sigma \cup V)^+$

- **START** Add a new start variable S' and a rule $S' \to S$
- **BIN** Replace each rule $A \to xu$ with the rules $A \to xA_1$ and $A_1 \to u$ and repeat until the RHS of every rule has length at most two
- **DEL-ε** For each rule of the form $A \to \varepsilon$ other than $S' \to \varepsilon$ remove $A \to \varepsilon$ and update all rules with A in the RHS
  - $B \to A$. Add rule $B \to \varepsilon$ unless $B \to \varepsilon$ has already been removed

Proof continued

In the following $x \in V \cup \Sigma$ and $u \in (\Sigma \cup V)^+$

START Add a new start variable S' and a rule $S' \to S$

BIN Replace each rule $A \to xu$ with the rules $A \to xA_1$ and $A_1 \to u$ and repeat until the RHS of every rule has length at most two

DEL-ε For each rule of the form $A \to \varepsilon$ other than $S' \to \varepsilon$ remove $A \to \varepsilon$ and update all rules with A in the RHS

- $B \to A$. Add rule $B \to \varepsilon$ unless $B \to \varepsilon$ has already been removed
- $B \to AA$. Add rule $B \to A$ and if $B \to \varepsilon$ has not already been removed, add it

Proof continued

In the following $x \in V \cup \Sigma$ and $u \in (\Sigma \cup V)^+$

START Add a new start variable S' and a rule $S' \to S$

BIN Replace each rule $A \to xu$ with the rules $A \to xA_1$ and $A_1 \to u$ and repeat until the RHS of every rule has length at most two

DEL-ε For each rule of the form $A \to \varepsilon$ other than $S' \to \varepsilon$ remove $A \to \varepsilon$ and update all rules with A in the RHS

- $B \to A$. Add rule $B \to \varepsilon$ unless $B \to \varepsilon$ has already been removed
- $B \to AA$. Add rule $B \to A$ and if $B \to \varepsilon$ has not already been removed, add it
- $B \to xA$ or $B \to Ax$. Add rule $B \to x$

Proof continued

In the following $x \in V \cup \Sigma$ and $u \in (\Sigma \cup V)^+$

START Add a new start variable S' and a rule $S' \to S$

BIN Replace each rule $A \to xu$ with the rules $A \to xA_1$ and $A_1 \to u$ and repeat until the RHS of every rule has length at most two

DEL-ε For each rule of the form $A \to \varepsilon$ other than $S' \to \varepsilon$ remove $A \to \varepsilon$ and update all rules with A in the RHS

- $B \to A$. Add rule $B \to \varepsilon$ unless $B \to \varepsilon$ has already been removed
- $B \to AA$. Add rule $B \to A$ and if $B \to \varepsilon$ has not already been removed, add it
- $B \to xA$ or $B \to Ax$. Add rule $B \to x$

UNIT For each rule $A \to B$, remove it and add rules $A \to u$ for each $B \to u$ unless $A \to u$ is a unit rule already removed

Proof continued

In the following $x \in V \cup \Sigma$ and $u \in (\Sigma \cup V)^+$

START Add a new start variable S' and a rule $S' \to S$

BIN Replace each rule $A \to xu$ with the rules $A \to xA_1$ and $A_1 \to u$ and repeat until the RHS of every rule has length at most two

DEL-ε For each rule of the form $A \to \varepsilon$ other than $S' \to \varepsilon$ remove $A \to \varepsilon$ and update all rules with A in the RHS

- $B \to A$. Add rule $B \to \varepsilon$ unless $B \to \varepsilon$ has already been removed
- $B \to AA$. Add rule $B \to A$ and if $B \to \varepsilon$ has not already been removed, add it
- $B \to xA$ or $B \to Ax$. Add rule $B \to x$

UNIT For each rule $A \to B$, remove it and add rules $A \to u$ for each $B \to u$ unless $A \to u$ is a unit rule already removed

TERM For each $t \in \Sigma$, add a new variable T and a rule $T \to t$; replace each t in the RHS of nonunit rules with T

Proof continued

In the following $x \in V \cup \Sigma$ and $u \in (\Sigma \cup V)^+$

- **START** Add a new start variable S' and a rule $S' \to S$
- **BIN** Replace each rule $A \to xu$ with the rules $A \to xA_1$ and $A_1 \to u$ and repeat until the RHS of every rule has length at most two
- **DEL-ε** For each rule of the form $A \to \varepsilon$ other than $S' \to \varepsilon$ remove $A \to \varepsilon$ and update all rules with A in the RHS
  - $B \to A$. Add rule $B \to \varepsilon$ unless $B \to \varepsilon$ has already been removed
  - $B \to AA$. Add rule $B \to A$ and if $B \to \varepsilon$ has not already been removed, add it
  - $B \to xA$ or $B \to Ax$. Add rule $B \to x$
- **UNIT** For each rule $A \to B$, remove it and add rules $A \to u$ for each $B \to u$ unless $A \to u$ is a unit rule already removed
- **TERM** For each $t \in \Sigma$, add a new variable T and a rule $T \to t$; replace each t in the RHS of nonunit rules with T

Each of the five steps preserves the language generated by the grammar so $L(G') = A$. □

Example

Convert to CNF

$$A \to BAB \mid B \mid \varepsilon$$

$$B \to 00 \mid \varepsilon$$

START:

Example

Convert to CNF

$$A \to BAB \mid B \mid \varepsilon$$

$$B \to 00 \mid \varepsilon$$

START:

$$S \to A$$

$$A \to BAB \mid B \mid \varepsilon$$

$$B \to 00 \mid \varepsilon$$

Example

Convert to CNF

$A \to BAB \mid B \mid \varepsilon$

$B \to 00 \mid \varepsilon$

START:

$S \to A$

$A \to BAB \mid B \mid \varepsilon$

$B \to 00 \mid \varepsilon$

BIN: Replace $A \to BAB$:

Example

Convert to CNF

$A \to BAB \mid B \mid \varepsilon$

$B \to 00 \mid \varepsilon$

START:

$S \to A$

$A \to BAB \mid B \mid \varepsilon$

$B \to 00 \mid \varepsilon$

BIN: Replace $A \to BAB$:

$S \to A$

$A \to BA_1 \mid B \mid \varepsilon$

$B \to 00 \mid \varepsilon$

$A_1 \to AB$

Example

Convert to CNF

$$A \to BAB \mid B \mid \varepsilon$$

$$B \to 00 \mid \varepsilon$$

START:

$$S \to A$$

$$A \to BAB \mid B \mid \varepsilon$$

$$B \to 00 \mid \varepsilon$$

BIN: Replace $A \to BAB$:

$$S \to A$$

$$A \to BA_1 \mid B \mid \varepsilon$$

$$B \to 00 \mid \varepsilon$$

$$A_1 \to AB$$

DEL-ε: Remove $A \to \varepsilon$:

Example

Convert to CNF

$$A \to BAB \mid B \mid \varepsilon$$

$$B \to 00 \mid \varepsilon$$

START:

$$S \to A$$

$$A \to BAB \mid B \mid \varepsilon$$

$$B \to 00 \mid \varepsilon$$

BIN: Replace $A \to BAB$:

$$S \to A$$

$$A \to BA_1 \mid B \mid \varepsilon$$

$$B \to 00 \mid \varepsilon$$

$$A_1 \to AB$$

DEL-ε: Remove $A \to \varepsilon$:

$$S \to A \mid \varepsilon$$

$$A \to BA_1 \mid B$$

$$B \to 00 \mid \varepsilon$$

$$A_1 \to AB \mid B$$

Example

Convert to CNF

$A \to BAB \mid B \mid \varepsilon$

$B \to 00 \mid \varepsilon$

START:

$S \to A$

$A \to BAB \mid B \mid \varepsilon$

$B \to 00 \mid \varepsilon$

BIN: Replace $A \to BAB$:

$S \to A$

$A \to BA_1 \mid B \mid \varepsilon$

$B \to 00 \mid \varepsilon$

$A_1 \to AB$

DEL-ε: Remove $A \to \varepsilon$:

$S \to A \mid \varepsilon$

$A \to BA_1 \mid B$

$B \to 00 \mid \varepsilon$

$A_1 \to AB \mid B$

Remove $B \to \varepsilon$:

Example

Convert to CNF

$$A \to BAB \mid B \mid \varepsilon$$

$$B \to 00 \mid \varepsilon$$

START:

$$S \to A$$

$$A \to BAB \mid B \mid \varepsilon$$

$$B \to 00 \mid \varepsilon$$

BIN: Replace $A \to BAB$:

$$S \to A$$

$$A \to BA_1 \mid B \mid \varepsilon$$

$$B \to 00 \mid \varepsilon$$

$$A_1 \to AB$$

DEL-ε: Remove $A \to \varepsilon$:

$$S \to A \mid \varepsilon$$

$$A \to BA_1 \mid B$$

$$B \to 00 \mid \varepsilon$$

$$A_1 \to AB \mid B$$

Remove $B \to \varepsilon$:

$$S \to A \mid \varepsilon$$

$$A \to BA_1 \mid B \mid A_1$$

$$B \to 00$$

$$A_1 \to AB \mid B \mid A \mid \varepsilon$$

Don't add $A \to \varepsilon$ because we already removed it

Example

Convert to CNF

$A \to BAB \mid B \mid \varepsilon$

$B \to 00 \mid \varepsilon$

START:

$S \to A$

$A \to BAB \mid B \mid \varepsilon$

$B \to 00 \mid \varepsilon$

BIN: Replace $A \to BAB$:

$S \to A$

$A \to BA_1 \mid B \mid \varepsilon$

$B \to 00 \mid \varepsilon$

$A_1 \to AB$

DEL-ε: Remove $A \to \varepsilon$:

$S \to A \mid \varepsilon$

$A \to BA_1 \mid B$

$B \to 00 \mid \varepsilon$

$A_1 \to AB \mid B$

Remove $B \to \varepsilon$:

$S \to A \mid \varepsilon$

$A \to BA_1 \mid B \mid A_1$

$B \to 00$

$A_1 \to AB \mid B \mid A \mid \varepsilon$

Don't add $A \to \varepsilon$ because we already removed it

Remove $A_1 \to \varepsilon$:

Example

Convert to CNF

$A \to BAB \mid B \mid \varepsilon$

$B \to 00 \mid \varepsilon$

START:

$S \to A$

$A \to BAB \mid B \mid \varepsilon$

$B \to 00 \mid \varepsilon$

BIN: Replace $A \to BAB$:

$S \to A$

$A \to BA_1 \mid B \mid \varepsilon$

$B \to 00 \mid \varepsilon$

$A_1 \to AB$

DEL-ε: Remove $A \to \varepsilon$:

$S \to A \mid \varepsilon$

$A \to BA_1 \mid B$

$B \to 00 \mid \varepsilon$

$A_1 \to AB \mid B$

Remove $B \to \varepsilon$:

$S \to A \mid \varepsilon$

$A \to BA_1 \mid B \mid A_1$

$B \to 00$

$A_1 \to AB \mid B \mid A \mid \varepsilon$

Don't add $A \to \varepsilon$ because we already removed it

Remove $A_1 \to \varepsilon$:

$S \to A \mid \varepsilon$

$A \to BA_1 \mid B \mid A_1$

$B \to 00$

$A_1 \to AB \mid B \mid A$

Don't add $A \to \varepsilon$ because we already removed it

Example

Convert to CNF

$A \to BAB \mid B \mid \varepsilon$

$B \to 00 \mid \varepsilon$

START:

$S \to A$

$A \to BAB \mid B \mid \varepsilon$

$B \to 00 \mid \varepsilon$

BIN: Replace $A \to BAB$:

$S \to A$

$A \to BA_1 \mid B \mid \varepsilon$

$B \to 00 \mid \varepsilon$

$A_1 \to AB$

DEL-ε: Remove $A \to \varepsilon$:

$S \to A \mid \varepsilon$

$A \to BA_1 \mid B$

$B \to 00 \mid \varepsilon$

$A_1 \to AB \mid B$

Remove $B \to \varepsilon$:

$S \to A \mid \varepsilon$

$A \to BA_1 \mid B \mid A_1$

$B \to 00$

$A_1 \to AB \mid B \mid A \mid \varepsilon$

Don't add $A \to \varepsilon$ because we already removed it

Remove $A_1 \to \varepsilon$:

$S \to A \mid \varepsilon$

$A \to BA_1 \mid B \mid A_1$

$B \to 00$

$A_1 \to AB \mid B \mid A$

Don't add $A \to \varepsilon$ because we already removed it

UNIT: Remove $S \to A$

Example

Convert to CNF

$A \to BAB \mid B \mid \varepsilon$

$B \to 00 \mid \varepsilon$

START:

$S \to A$

$A \to BAB \mid B \mid \varepsilon$

$B \to 00 \mid \varepsilon$

BIN: Replace $A \to BAB$:

$S \to A$

$A \to BA_1 \mid B \mid \varepsilon$

$B \to 00 \mid \varepsilon$

$A_1 \to AB$

DEL-ε: Remove $A \to \varepsilon$:

$S \to A \mid \varepsilon$

$A \to BA_1 \mid B$

$B \to 00 \mid \varepsilon$

$A_1 \to AB \mid B$

Remove $B \to \varepsilon$:

$S \to A \mid \varepsilon$

$A \to BA_1 \mid B \mid A_1$

$B \to 00$

$A_1 \to AB \mid B \mid A \mid \varepsilon$

Don't add $A \to \varepsilon$ because we already removed it

Remove $A_1 \to \varepsilon$:

$S \to A \mid \varepsilon$

$A \to BA_1 \mid B \mid A_1$

$B \to 00$

$A_1 \to AB \mid B \mid A$

Don't add $A \to \varepsilon$ because we already removed it

UNIT: Remove $S \to A$

$S \to BA_1 \mid B \mid A_1 \mid \varepsilon$

$A \to BA_1 \mid B \mid A_1$

$B \to 00$

$A_1 \to AB \mid B \mid A$

Example continued

From previous slide

$$\begin{aligned} S &\to BA_1 \mid B \mid A_1 \mid \varepsilon \\ A &\to BA_1 \mid B \mid A_1 \\ B &\to 00 \\ A_1 &\to AB \mid B \mid A \end{aligned}$$

Example continued

From previous slide

$$
\begin{aligned}
S &\to BA_1 \mid B \mid A_1 \mid \varepsilon \\
A &\to BA_1 \mid B \mid A_1 \\
B &\to 00 \\
A_1 &\to AB \mid B \mid A
\end{aligned}
$$

Remove $S \to B$

Example continued

From previous slide

$$S \to BA_1 \mid B \mid A_1 \mid \varepsilon$$
$$A \to BA_1 \mid B \mid A_1$$
$$B \to 00$$
$$A_1 \to AB \mid B \mid A$$

Remove $S \to B$

$$S \to BA_1 \mid A_1 \mid \varepsilon \mid 00$$
$$A \to BA_1 \mid B \mid A_1$$
$$B \to 00$$
$$A_1 \to AB \mid B \mid A$$

Example continued

From previous slide

$$S \to BA_1 \mid B \mid A_1 \mid \varepsilon$$
$$A \to BA_1 \mid B \mid A_1$$
$$B \to 00$$
$$A_1 \to AB \mid B \mid A$$

Remove $S \to B$

$$S \to BA_1 \mid A_1 \mid \varepsilon \mid 00$$
$$A \to BA_1 \mid B \mid A_1$$
$$B \to 00$$
$$A_1 \to AB \mid B \mid A$$

Remove $S \to A_1$

Example continued

From previous slide

$$S \to BA_1 \mid B \mid A_1 \mid \varepsilon$$
$$A \to BA_1 \mid B \mid A_1$$
$$B \to 00$$
$$A_1 \to AB \mid B \mid A$$

Remove $S \to A_1$

$$S \to BA_1 \mid \varepsilon \mid 00 \mid AB$$
$$A \to BA_1 \mid B \mid A_1$$
$$B \to 00$$
$$A_1 \to AB \mid B \mid A$$

Remove $S \to B$

$$S \to BA_1 \mid A_1 \mid \varepsilon \mid 00$$
$$A \to BA_1 \mid B \mid A_1$$
$$B \to 00$$
$$A_1 \to AB \mid B \mid A$$

Don't add $S \to B$ or $S \to A$ because we removed them

Example continued

From previous slide

$$S \to BA_1 \mid B \mid A_1 \mid \varepsilon$$
$$A \to BA_1 \mid B \mid A_1$$
$$B \to 00$$
$$A_1 \to AB \mid B \mid A$$

Remove $S \to B$

$$S \to BA_1 \mid A_1 \mid \varepsilon \mid 00$$
$$A \to BA_1 \mid B \mid A_1$$
$$B \to 00$$
$$A_1 \to AB \mid B \mid A$$

Remove $S \to A_1$

$$S \to BA_1 \mid \varepsilon \mid 00 \mid AB$$
$$A \to BA_1 \mid B \mid A_1$$
$$B \to 00$$
$$A_1 \to AB \mid B \mid A$$

Don't add $S \to B$ or $S \to A$ because we removed them

Remove $A \to B$

Example continued

From previous slide

$$S \to BA_1 \mid B \mid A_1 \mid \varepsilon$$
$$A \to BA_1 \mid B \mid A_1$$
$$B \to 00$$
$$A_1 \to AB \mid B \mid A$$

Remove $S \to B$

$$S \to BA_1 \mid A_1 \mid \varepsilon \mid 00$$
$$A \to BA_1 \mid B \mid A_1$$
$$B \to 00$$
$$A_1 \to AB \mid B \mid A$$

Remove $S \to A_1$

$$S \to BA_1 \mid \varepsilon \mid 00 \mid AB$$
$$A \to BA_1 \mid B \mid A_1$$
$$B \to 00$$
$$A_1 \to AB \mid B \mid A$$

Don't add $S \to B$ or $S \to A$ because we removed them

Remove $A \to B$

$$S \to BA_1 \mid \varepsilon \mid 00 \mid AB$$
$$A \to BA_1 \mid A_1 \mid 00$$
$$B \to 00$$
$$A_1 \to AB \mid B \mid A$$

Example continued

From previous slide

$$S \to BA_1 \mid B \mid A_1 \mid \varepsilon$$
$$A \to BA_1 \mid B \mid A_1$$
$$B \to 00$$
$$A_1 \to AB \mid B \mid A$$

Remove $S \to B$

$$S \to BA_1 \mid A_1 \mid \varepsilon \mid 00$$
$$A \to BA_1 \mid B \mid A_1$$
$$B \to 00$$
$$A_1 \to AB \mid B \mid A$$

Remove $S \to A_1$

$$S \to BA_1 \mid \varepsilon \mid 00 \mid AB$$
$$A \to BA_1 \mid B \mid A_1$$
$$B \to 00$$
$$A_1 \to AB \mid B \mid A$$

Don't add $S \to B$ or $S \to A$ because we removed them

Remove $A \to B$

$$S \to BA_1 \mid \varepsilon \mid 00 \mid AB$$
$$A \to BA_1 \mid A_1 \mid 00$$
$$B \to 00$$
$$A_1 \to AB \mid B \mid A$$

Remove $A \to A_1$

Example continued

From previous slide

$$S \to BA_1 \mid B \mid A_1 \mid \varepsilon$$
$$A \to BA_1 \mid B \mid A_1$$
$$B \to 00$$
$$A_1 \to AB \mid B \mid A$$

Remove $S \to A_1$

$$S \to BA_1 \mid \varepsilon \mid 00 \mid AB$$
$$A \to BA_1 \mid B \mid A_1$$
$$B \to 00$$
$$A_1 \to AB \mid B \mid A$$

Remove $A \to A_1$

$$S \to BA_1 \mid \varepsilon \mid 00 \mid AB$$
$$A \to BA_1 \mid 00 \mid AB$$
$$B \to 00$$
$$A_1 \to AB \mid B \mid A$$

Remove $S \to B$

$$S \to BA_1 \mid A_1 \mid \varepsilon \mid 00$$
$$A \to BA_1 \mid B \mid A_1$$
$$B \to 00$$
$$A_1 \to AB \mid B \mid A$$

Don't add $S \to B$ or $S \to A$ because we removed them

Remove $A \to B$

$$S \to BA_1 \mid \varepsilon \mid 00 \mid AB$$
$$A \to BA_1 \mid A_1 \mid 00$$
$$B \to 00$$
$$A_1 \to AB \mid B \mid A$$

Don't add $A \to B$ because we removed it

Don't add $A \to A$ because it's useless

Example continued

From previous slide

$$S \to BA_1 \mid B \mid A_1 \mid \varepsilon$$
$$A \to BA_1 \mid B \mid A_1$$
$$B \to 00$$
$$A_1 \to AB \mid B \mid A$$

Remove $S \to A_1$

$$S \to BA_1 \mid \varepsilon \mid 00 \mid AB$$
$$A \to BA_1 \mid B \mid A_1$$
$$B \to 00$$
$$A_1 \to AB \mid B \mid A$$

Remove $A \to A_1$

$$S \to BA_1 \mid \varepsilon \mid 00 \mid AB$$
$$A \to BA_1 \mid 00 \mid AB$$
$$B \to 00$$
$$A_1 \to AB \mid B \mid A$$

Remove $S \to B$

$$S \to BA_1 \mid A_1 \mid \varepsilon \mid 00$$
$$A \to BA_1 \mid B \mid A_1$$
$$B \to 00$$
$$A_1 \to AB \mid B \mid A$$

Don't add $S \to B$ or $S \to A$ because we removed them

Remove $A \to B$

$$S \to BA_1 \mid \varepsilon \mid 00 \mid AB$$
$$A \to BA_1 \mid A_1 \mid 00$$
$$B \to 00$$
$$A_1 \to AB \mid B \mid A$$

Don't add $A \to B$ because we removed it

Don't add $A \to A$ because it's useless

Remove $A_1 \to B$

Example continued

From previous slide

$$S \to BA_1 \mid B \mid A_1 \mid \varepsilon$$
$$A \to BA_1 \mid B \mid A_1$$
$$B \to 00$$
$$A_1 \to AB \mid B \mid A$$

Remove $S \to B$

$$S \to BA_1 \mid A_1 \mid \varepsilon \mid 00$$
$$A \to BA_1 \mid B \mid A_1$$
$$B \to 00$$
$$A_1 \to AB \mid B \mid A$$

Remove $S \to A_1$

$$S \to BA_1 \mid \varepsilon \mid 00 \mid AB$$
$$A \to BA_1 \mid B \mid A_1$$
$$B \to 00$$
$$A_1 \to AB \mid B \mid A$$

Don't add $S \to B$ or $S \to A$ because we removed them

Remove $A \to B$

$$S \to BA_1 \mid \varepsilon \mid 00 \mid AB$$
$$A \to BA_1 \mid A_1 \mid 00$$
$$B \to 00$$
$$A_1 \to AB \mid B \mid A$$

Remove $A \to A_1$

$$S \to BA_1 \mid \varepsilon \mid 00 \mid AB$$
$$A \to BA_1 \mid 00 \mid AB$$
$$B \to 00$$
$$A_1 \to AB \mid B \mid A$$

Don't add $A \to B$ because we removed it

Don't add $A \to A$ because it's useless

Remove $A_1 \to B$

$$S \to BA_1 \mid \varepsilon \mid 00 \mid AB$$
$$A \to BA_1 \mid 00 \mid AB$$
$$B \to 00$$
$$A_1 \to AB \mid A \mid 00$$

Example continued

Copied from the previous slide

$$
\begin{aligned}
S &\rightarrow BA_1 \mid \varepsilon \mid 00 \mid AB \\
A &\rightarrow BA_1 \mid 00 \mid AB \\
B &\rightarrow 00 \\
A_1 &\rightarrow AB \mid A \mid 00
\end{aligned}
$$

Remove $A_1 \rightarrow A$

Example continued

Copied from the previous slide

$$S \to BA_1 \mid \varepsilon \mid 00 \mid AB$$

$$A \to BA_1 \mid 00 \mid AB$$

$$B \to 00$$

$$A_1 \to AB \mid A \mid 00$$

Remove $A_1 \to A$

$$S \to BA_1 \mid \varepsilon \mid 00 \mid AB$$

$$A \to BA_1 \mid 00 \mid AB$$

$$B \to 00$$

$$A_1 \to AB \mid 00 \mid BA_1$$

Example continued

Copied from the previous slide

$$S \to BA_1 \mid \varepsilon \mid 00 \mid AB$$

$$A \to BA_1 \mid 00 \mid AB$$

$$B \to 00$$

$$A_1 \to AB \mid A \mid 00$$

Remove $A_1 \to A$

$$S \to BA_1 \mid \varepsilon \mid 00 \mid AB$$

$$A \to BA_1 \mid 00 \mid AB$$

$$B \to 00$$

$$A_1 \to AB \mid 00 \mid BA_1$$

TERM: Add $Z \to 0$

Example continued

Copied from the previous slide

$$S \to BA_1 \mid \varepsilon \mid 00 \mid AB$$
$$A \to BA_1 \mid 00 \mid AB$$
$$B \to 00$$
$$A_1 \to AB \mid A \mid 00$$

Remove $A_1 \to A$

$$S \to BA_1 \mid \varepsilon \mid 00 \mid AB$$
$$A \to BA_1 \mid 00 \mid AB$$
$$B \to 00$$
$$A_1 \to AB \mid 00 \mid BA_1$$

TERM: Add $Z \to 0$

$$S \to BA_1 \mid \varepsilon \mid ZZ \mid AB$$
$$A \to BA_1 \mid ZZ \mid AB$$
$$B \to ZZ$$
$$A_1 \to AB \mid ZZ \mid BA_1$$
$$Z \to 0$$

Caution

Sipser gives a different procedure

1. START
2. DEL-ε
3. UNIT
4. BIN
5. TERM

This procedure works but can lead to an exponential blow up in the number of rules!

In general, if DEL-ε comes before BIN, then $|G'|$ is $O(2^{2^{|G|}})$;
if BIN comes before DEL-ε, then $|G'|$ is $O(|G|^2)$

UNIT is responsible for the quadratic blow up

So use whichever procedure you'd like, but Sipser's can be *very* bad
(Sipser's is bad if you have long rules with lots of variables with ε-rules)

Example blow up

$$
\begin{aligned}
A &\to BCDEEDCB \mid CBEDDEBC \\
B &\to 0 \mid \varepsilon \\
C &\to 1 \mid \varepsilon \\
D &\to 2 \mid \varepsilon \\
E &\to 3 \mid \varepsilon
\end{aligned}
$$

has five variables and 10 rules

Converting using START, BIN, DEL-ε, UNIT, TERM gives a CFG with 18 variables and 125 rules

Example blow up

$$A \to BCDEEDCB \mid CBEDDEBC$$
$$B \to 0 \mid \varepsilon$$
$$C \to 1 \mid \varepsilon$$
$$D \to 2 \mid \varepsilon$$
$$E \to 3 \mid \varepsilon$$

has five variables and 10 rules

Converting using START, BIN, DEL-ε, UNIT, TERM gives a CFG with 18 variables and 125 rules

Converting using START, DEL-ε, UNIT, BIN, TERM gives a CFG with 1394 variables and 1953 rules

Prefix

Recall $\textsc{Prefix}(L) = \{w \mid \text{for some } x \in \Sigma^*, wx \in L\}$

Theorem

The class of context-free languages is closed under $\textsc{Prefix}$*.*

Prefix

Recall $\textsc{Prefix}(L) = \{w \mid \text{for some } x \in \Sigma^*,\ wx \in L\}$

Theorem

The class of context-free languages is closed under $\textsc{Prefix}$*.*

Proof idea

Consider the language $\{w\#w^{\mathcal{R}} \mid w \in \{\mathtt{a}, \mathtt{b}\}^*\}$ generated by

$$T \to \mathtt{a}T\mathtt{a} \mid \mathtt{b}T\mathtt{b} \mid \#$$

Prefix

Recall $\textsc{Prefix}(L) = \{w \mid \text{for some } x \in \Sigma^*, wx \in L\}$

Theorem

The class of context-free languages is closed under $\textsc{Prefix}$.

Proof idea

Consider the language $\{w\#w^{\mathcal{R}} \mid w \in \{\mathtt{a}, \mathtt{b}\}^*\}$ generated by

$$T \to \mathtt{a}T\mathtt{a} \mid \mathtt{b}T\mathtt{b} \mid \#$$

Let's convert to CNF

Prefix

Recall $\textsc{Prefix}(L) = \{w \mid \text{for some } x \in \Sigma^*, wx \in L\}$

Theorem

The class of context-free languages is closed under $\textsc{Prefix}$.

Proof idea

Consider the language $\{w\#w^{\mathcal{R}} \mid w \in \{\mathtt{a}, \mathtt{b}\}^*\}$ generated by

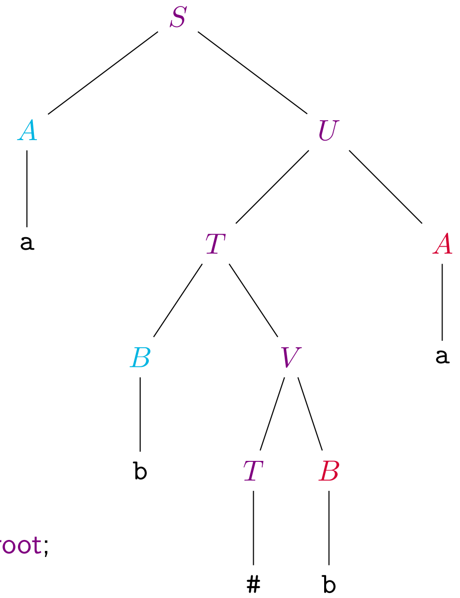
$$T \to \mathtt{a}T\mathtt{a} \mid \mathtt{b}T\mathtt{b} \mid \#$$

Let's convert to CNF

$$S \to AU \mid BV \mid \#$$
$$T \to AU \mid BV \mid \#$$
$$U \to TA$$
$$V \to TB$$
$$A \to \mathtt{a}$$
$$B \to \mathtt{b}$$

Derivation of ab#ba

$$
\begin{aligned}
S &\Rightarrow AU \\
&\Rightarrow \mathtt{a}U \\
&\Rightarrow \mathtt{a}TA \\
&\Rightarrow \mathtt{a}BVA \\
&\Rightarrow \mathtt{ab}VA \\
&\Rightarrow \mathtt{ab}TBA \\
&\Rightarrow \mathtt{ab\#}BA \\
&\Rightarrow \mathtt{ab\#b}A \\
&\Rightarrow \mathtt{ab\#ba}
\end{aligned}
$$

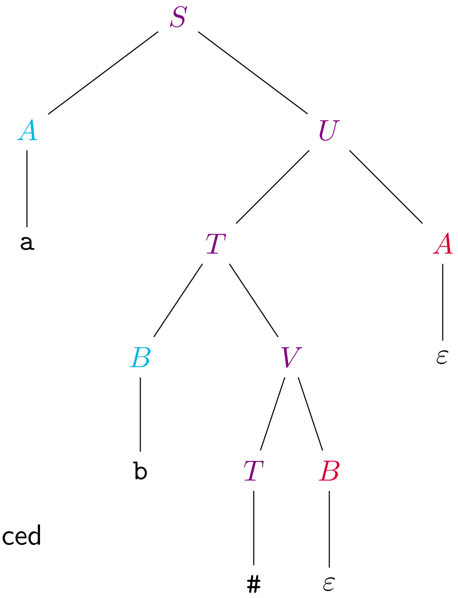


The prefix ab# includes
– all terminals from subtrees with a blue root;
– some terminals from subtrees with a violet root;
– no terminals from subtrees with a red root

Desired derivation for the prefix

We would like a derivation like this

$$
\begin{aligned}
S &\Rightarrow AU \\
&\Rightarrow \texttt{a}U \\
&\Rightarrow \texttt{a}TA \\
&\Rightarrow \texttt{a}BVA \\
&\Rightarrow \texttt{ab}VA \\
&\Rightarrow \texttt{ab}TBA \\
&\Rightarrow \texttt{ab\#}BA \\
&\Rightarrow \texttt{ab\#}\varepsilon A \\
&\Rightarrow \texttt{ab\#}\varepsilon\varepsilon
\end{aligned}
$$



Everything left of the violet path is produced
Everything right of the violet path becomes ε
The leaf connected to the violet path is produced

The proof idea

The violet path corresponds to the point where we "split" the prefix from the remainder of the string

We want to construct a CFG that keeps track of whether a given variable in the derivation is

L left of the split,

S part of the split, or

R right of the split

We can construct a new CFG whose variables are $\langle A, L\rangle$, $\langle A, S\rangle$, or $\langle A, R\rangle$ where A is a variable in the original CFG

We have to deal with the three types of rules

- $S \to \varepsilon$
- $A \to BC$
- $A \to t$

and produce new rules corresponding to the variable on the LHS being left of, right of, or on the split

Proof

If $L = \varnothing$, then $\textsc{Prefix}(L) = \varnothing$ which is CF.

Otherwise, let L be CF and generated by the CFG $G = (V, \Sigma, R, S)$ in CNF.

Construct a new CFG (not in CNF) $G' = (V', \Sigma, R', S')$ where

$$V = \{\langle A, D\rangle \mid A \in V \text{ and } D \in \{L, S, R\}\}$$

$$S' = \langle S, S\rangle$$

Now we just need to specify R'. We'll start with $R' = \varnothing$ and add rules to it

Proof continued

Since L is nonempty, $\varepsilon \in \textsc{Prefix}(L)$ so add the rule $\langle S, S\rangle \to \varepsilon$ to R'

For each rule of the form $A \to BC$ in R, add the following rules to R'

$\langle A, L\rangle \to \langle B, L\rangle\langle C, L\rangle$	left of the split
$\langle A, S\rangle \to \langle B, L\rangle\langle C, S\rangle \mid \langle B, S\rangle\langle C, R\rangle$	one of B or C is on the split
$\langle A, R\rangle \to \langle B, R\rangle\langle C, R\rangle$	right of the split

For each rule of the form $A \to t$ in R, add the following rules to R'

$\langle A, L\rangle \to t$

$\langle A, S\rangle \to t$

$\langle A, R\rangle \to \varepsilon$

Proof continued

For each $w = w_1 w_2 \cdots w_n \in L$, $S \stackrel{*}{\Rightarrow} A_1 A_2 \cdots A_n$ where $A_i \Rightarrow w_i$
By construction,

$$\begin{aligned}\langle S, S\rangle &\stackrel{*}{\Rightarrow} \langle A_1, L\rangle \cdots \langle A_{i-1}, L\rangle \langle A_i, S\rangle \langle A_{i+1}, R\rangle \cdots \langle A_n, R\rangle \\ &\stackrel{*}{\Rightarrow} w_1 w_2 \cdots w_i\end{aligned}$$

for each $1 \le i \le n$

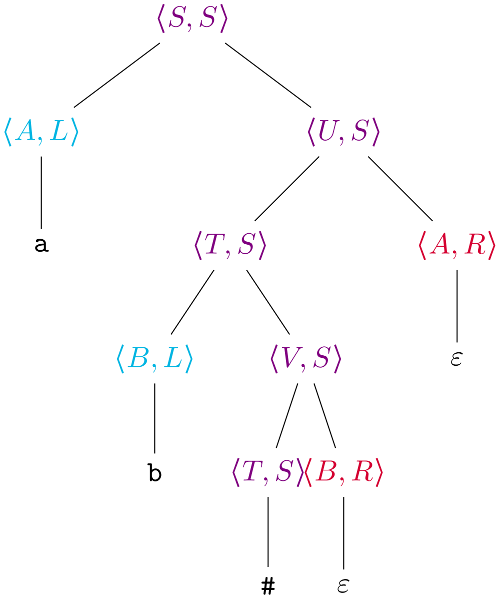
I.e., G' derives the prefix of every string in L

A similar argument works to show that if G' derives a string then it's a prefix of some string in L $\square$

Applying the construction

Deriving `ab#`

$$
\begin{aligned}
\langle S, S\rangle &\Rightarrow \langle A, L\rangle\langle U, S\rangle \\
&\Rightarrow \mathtt{a}\langle U, S\rangle \\
&\Rightarrow \mathtt{a}\langle T, S\rangle\langle A, R\rangle \\
&\Rightarrow \mathtt{a}\langle B, L\rangle\langle V, S\rangle\langle A, R\rangle \\
&\Rightarrow \mathtt{ab}\langle V, S\rangle\langle A, R\rangle \\
&\Rightarrow \mathtt{ab}\langle T, S\rangle\langle BA, R\rangle \\
&\Rightarrow \mathtt{ab\#}\langle B, R\rangle\langle A, R\rangle \\
&\Rightarrow \mathtt{ab\#}\langle A, R\rangle \\
&\Rightarrow \mathtt{ab\#}
\end{aligned}
$$



Similarities with regular expression

Proving things about

- Regular languages. Assume there exists a regular expression that generates the language and consider the six cases
- Context-free languages. Assume there exists a CFG that generates the language and consider the three types of rules