Finite automata and formal languages (DIT322, TMV028)

Nils Anders Danielsson

2020-02-27

Today

- Grammar transformations.
- Chomsky normal form.
- The pumping lemma for context-free languages.

Old exams

If you look at the sample solutions for the old exams, note that I have changed some notation:

- L_{L} used to be L^*.

Grammar transformations

Grammar transformations

- A number of transformations of grammars.
- Will be used for parsing (next lecture).
- I have taken some information and terminology from "To CNF or not to CNF? An Efficient Yet Presentable Version of the CYK Algorithm" by Lange and Leiß.

$\textsc{Bin}$

- Result: No production $A \to \alpha$ where $|\alpha| \geq 3$.
- Replace each production $A \to X_1 X_2 ... X_n$, where $n \geq 3$, with:

$$\begin{aligned} A &\to X_1 A_2 \\ A_2 &\to X_2 A_3 \\ &\vdots \\ A_{n-1} &\to X_{n-1} X_n \end{aligned}$$

  Here $A_2, ..., A_{n-1}$ are new nonterminals.
- $L(\textsc{Bin}(G)) = L(G)$.

Del

- Result: No “deletion rules”,
  i.e. productions of the form $A \to \varepsilon$.
- A nonterminal A is *nullable* if $A \Rightarrow^* \varepsilon$.

Del

An example:

- Replace the production $A \to \alpha B \beta C \gamma$, where B and C are the only nullable nonterminals, with

$$A \to \alpha B \beta C \gamma,$$
$$A \to \alpha \beta C \gamma,$$
$$A \to \alpha B \beta \gamma \text{ and, if } \alpha\beta\gamma \neq \varepsilon,$$
$$A \to \alpha \beta \gamma.$$

- The new productions are not deletion rules.
- If we do this for every production, then no nonterminal will be nullable, and $L(\mathrm{Del}(G), A) = L(G, A) \setminus \{ \varepsilon \}$.

DEL

$$L(\textsc{Del}(G)) = L(G) \setminus \{ \varepsilon \}.$$

Del

If Del is applied to the following grammar, how many productions does the resulting grammar contain?

$$(\{\, S, A \,\}, \{\, 0 \,\}, (S \to (SA)^{10} \mid \varepsilon, A \to 0), S)$$

Del

- The Del transformation can make the grammar much larger.
- If every production $A \to \alpha$ satisfies $|\alpha| \leq 2$, then the blowup is contained.
- Run Bin before Del.

Unit

- Result: No production of the form $A \to B$.
- (A, B) is a *unit pair* if $A = B$ or $A \to C_1 \to \cdots \to C_n \to B$ (where $n \in \mathbb{N}$).
- Include exactly the following productions:

$$\{A \to \alpha \mid (A, B) \text{ is a unit pair, } B \to \alpha \in P, \alpha \text{ is not a single nonterminal}\}$$

Unit

Example:

- Before:

$$
\begin{aligned}
A &\to 1 \mid B \\
B &\to 2 \mid C \\
C &\to AB
\end{aligned}
$$

- After:

$$
\begin{aligned}
A &\to 1 \mid 2 \mid AB \\
B &\to 2 \mid AB \\
C &\to AB
\end{aligned}
$$

$L(\textsc{Unit}(G)) = L(G)$.

Unit

The resulting grammar could be much larger than the original one:

$$
\begin{aligned}
A_1 &\to A_2 \mid 1 \\
A_2 &\to A_3 \mid 2 \\
A_3 &\to A_4 \mid 3 \\
&\vdots \\
A_n &\to A_1 \mid n
\end{aligned}
$$

Unit

The resulting grammar could be much larger than the original one:

$$
\begin{aligned}
A_1 &\to 1 \mid 2 \mid 3 \mid \ldots \mid n \\
A_2 &\to 1 \mid 2 \mid 3 \mid \ldots \mid n \\
A_3 &\to 1 \mid 2 \mid 3 \mid \ldots \mid n \\
&\vdots \\
A_n &\to 1 \mid 2 \mid 3 \mid \ldots \mid n
\end{aligned}
$$

Construct a grammar G for which $\textsc{Del}(\textsc{Unit}(G))$ contains a production of the form $A \to B$.

Construct a grammar G for which $\textsc{Del}(\textsc{Unit}(G))$ contains a production of the form $A \rightarrow B$.

Run $\textsc{Del}$ before $\textsc{Unit}$.

TERM

- Result: No terminals in productions $A \to \alpha$ where $|\alpha| \geq 2$.
- Find all terminals in such productions.
- For each such terminal b, add a new nonterminal B with a single production $B \to b$, and substitute B for b in every production $A \to \alpha$ where $|\alpha| \geq 2$.
- $L(\text{TERM}(G)) = L(G)$.

Bin/Term

- I have written $\textsc{Bin}(G)$ and $\textsc{Term}(G)$, as if $\textsc{Bin}$ and $\textsc{Term}$ were functions.
- However, these transformations are not functions, because the names of the new nonterminals are not uniquely specified.
- Below I will pretend that the transformations are functions.

Chomsky normal form

Chomsky normal form

- A context-free grammar is in *Chomsky normal form* if every production is of the form $A \to BC$ or $A \to a$.
- For any context-free grammar G the grammar $G' = \textsc{Term}(\textsc{Unit}(\textsc{Del}(\textsc{Bin}(G))))$ is in Chomsky normal form and satisfies $L(G') = L(G) \setminus \{ \varepsilon \}$.

Chomsky normal form

- A context-free grammar is in *Chomsky normal form* if every production is of the form $A \to BC$ or $A \to a$.
- For any context-free grammar G the grammar $G' = \textsc{Term}(\textsc{Unit}(\textsc{Del}(\textsc{Bin}(G))))$ is in Chomsky normal form and satisfies $L(G') = L(G) \setminus \{ \varepsilon \}$.

I dropped the text book's requirement that there should be no useless symbols.

Consider the grammar $G = (\{ S, A \}, \{ 0, 1 \}, P, S)$, where P is defined in the following way:

$$S \to 0A \mid S$$
$$A \to 1S \mid \varepsilon$$

- Is G ambiguous?
- Is $\textsc{Term}(\textsc{Unit}(\textsc{Del}(\textsc{Bin}(G))))$ ambiguous?

Consider the grammar $G = (\{ S, A \}, \{ 0, 1 \}, P, S)$, where P is defined in the following way:

$$S \to 0A \mid S$$
$$A \to 1S \mid \varepsilon$$

- Is G ambiguous?
- Is $\textsc{Term}(\textsc{Unit}(\textsc{Del}(\textsc{Bin}(G))))$ ambiguous?

If G is ambiguous, then $\textsc{Unit}(G)$ is sometimes ambiguous, sometimes not.

The pumping lemma

The pumping lemma for CFLs

For every context-free language L
over the alphabet Σ:

$$
\begin{aligned}
&\exists m \in \mathbb{N}.\\
&\quad \forall w \in L.\ |w| \geq m \Rightarrow\\
&\quad\quad \exists r, s, t, u, v \in \Sigma^*.\\
&\quad\quad\quad w = rstuv \wedge |stu| \leq m \wedge su \neq \varepsilon \wedge\\
&\quad\quad\quad \forall n \in \mathbb{N}.\ rs^n tu^n v \in L
\end{aligned}
$$

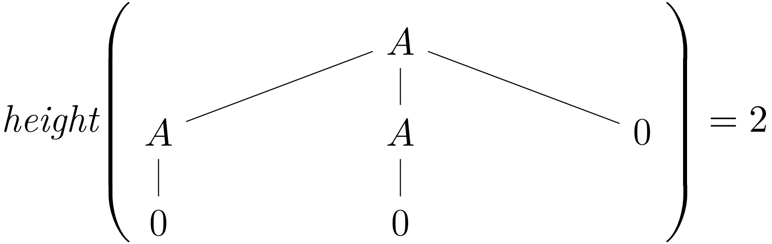
The pumping lemma for CFLs

For every context-free language L over the alphabet Σ:

$$\begin{aligned}
&\exists m \in \mathbb{N}.\\
&\quad \forall w \in L.\ |w| \geq m \Rightarrow\\
&\qquad \exists r, s, t, u, v \in \Sigma^*.\\
&\qquad\quad w = rstuv \wedge |stu| \leq m \wedge su \neq \varepsilon \wedge\\
&\qquad\quad \forall n \in \mathbb{N}.\ rs^n t u^n v \in L
\end{aligned}$$

Height

The height of a parse tree in $P(G, A)$ is the largest number of nonterminals encountered on any path from the root to a leaf.



Height

For context-free grammars in Chomsky normal form:

$$\forall p \in P(G, A).\ |yield(p)| \leq 2^{height(p)-1}$$

Height

For context-free grammars in Chomsky normal form:

$$\forall p \in P(G, A).\ |yield(p)| \leq 2^{height(p)-1}$$

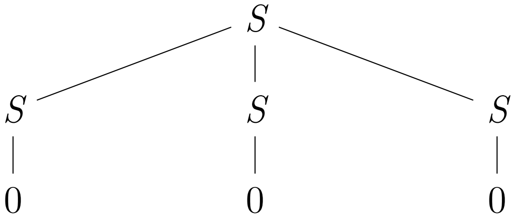
Proof: Exercise.

Height

Consider the following grammar and parse tree:

$$(\{\, S \,\}, \{\, 0 \,\}, (S \to SSS \mid 0), S)$$

$p =$



We have
$|yield(p)| = |000| = 3 \not\leq 2 = 2^{2-1} = 2^{height(p)-1}$.

The pumping lemma for CFLs

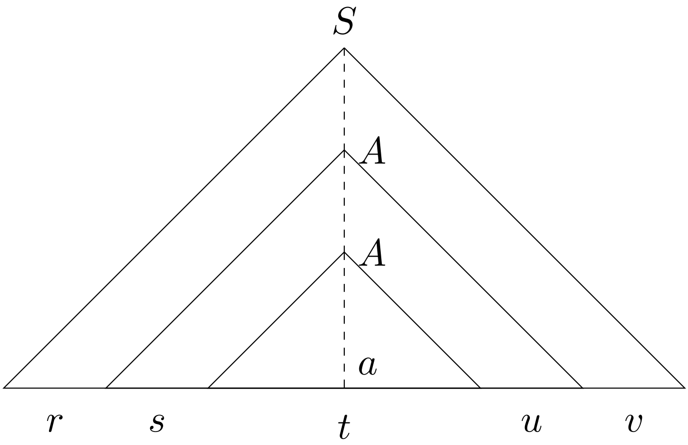
Proof sketch:

- Take any context-free grammar G for L.
- Let $G' = \textsc{Term}(\textsc{Unit}(\textsc{Del}(\textsc{Bin}(G))))$.
- If $G' = (N, \Sigma, P, S)$, let $m = 2^{|N|}$.
- Given a string $w \in L$ with $|w| \geq m$ we know that $w \neq \varepsilon$, so we have $w \in L \setminus \{ \varepsilon \} = L(G')$.

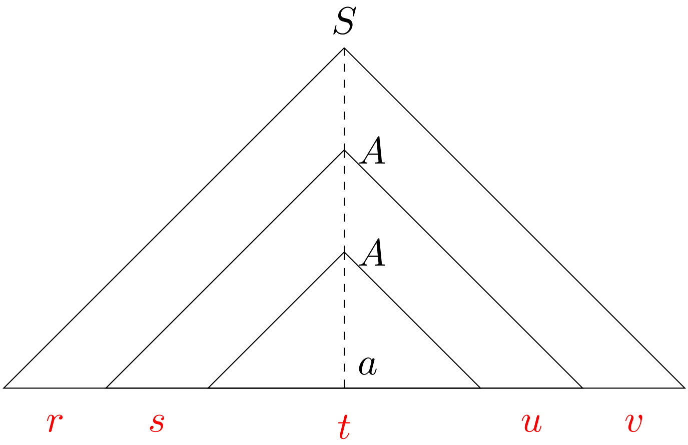
The pumping lemma for CFLs

- Take any parse tree p for w with respect to G'.
- We know that $2^{|N|} = m \leq |w| \leq 2^{height(p)-1}$, so $height(p) > |N|$.
- Take a path of maximal length from the root of p to a leaf.
- Such a path must contain at least $|N| + 1$ nonterminals.
- By the pigeonhole principle the path must contain two instances of the same nonterminal, at most $|N| + 1$ steps from the leaf.

The pumping lemma for CFLs

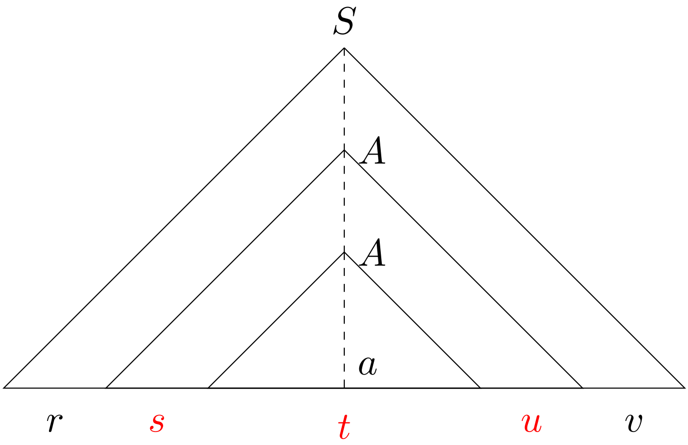


The pumping lemma for CFLs



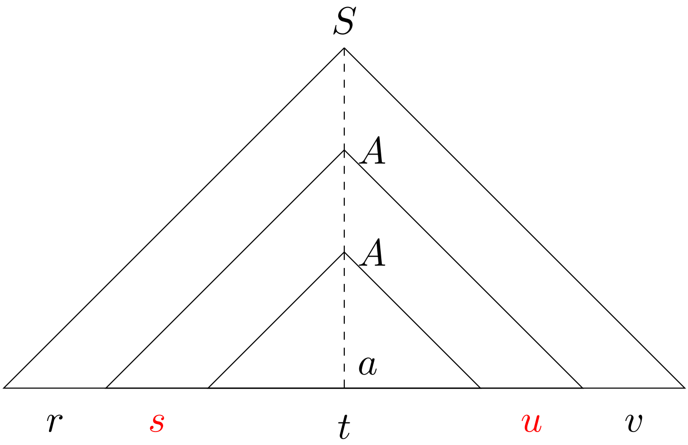
$w = rstuv$

The pumping lemma for CFLs



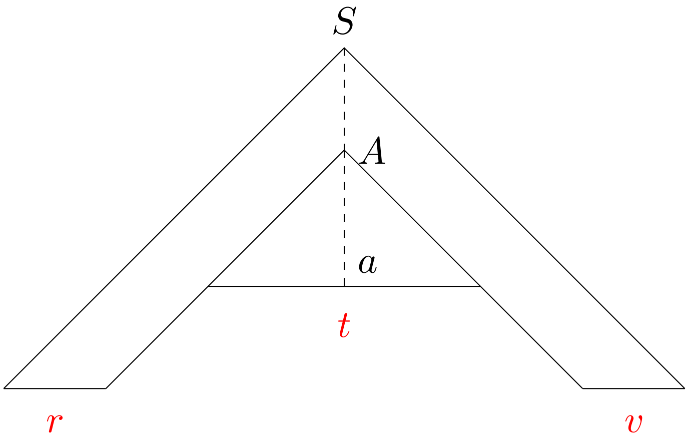
$$|stu| \leq 2^{(|N|+1)-1} = 2^{|N|} = m$$

The pumping lemma for CFLs



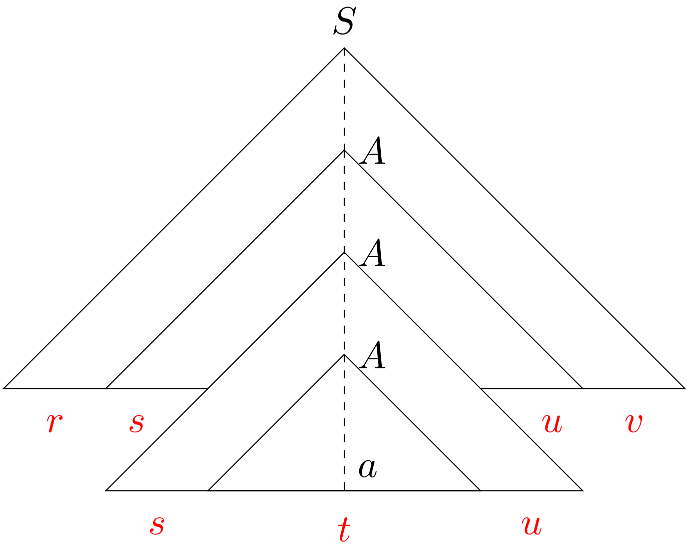
No nonterminal is nullable, $A \to BC \Rightarrow$
$s \neq \varepsilon \vee u \neq \varepsilon \Rightarrow su \neq \varepsilon$

The pumping lemma for CFLs



$rtv \in L(G') \subseteq L$

The pumping lemma for CFLs



$rs^2tu^2v \in L(G') \subseteq L$

The pumping lemma for CFLs

The language $L = \{ 0^n 1^n 2^n \mid n \in \mathbb{N} \}$ over $\Sigma = \{ 0, 1, 2 \}$ is not context-free. Proof sketch:

- Assume that L is context-free.
- Take the constant $m \in \mathbb{N}$ that we get from the pumping lemma.
- Consider the string $w = 0^m 1^m 2^m \in L$.
- Because $|w| \geq m$ we get some information:

$$
\begin{aligned}
&\exists r, s, t, u, v \in \Sigma^*. \\
&\quad w = rstuv \wedge |stu| \leq m \wedge su \neq \varepsilon \wedge \\
&\quad \forall n \in \mathbb{N}.\ rs^n tu^n v \in L
\end{aligned}
$$

The pumping lemma for CFLs

- Because $|w| \geq m$ we get some information:

$$\begin{aligned}&\exists r, s, t, u, v \in \Sigma^*.\\&\quad w = rstuv \wedge |stu| \leq m \wedge su \neq \varepsilon \wedge\\&\quad \forall n \in \mathbb{N}.\ rs^n tu^n v \in L\end{aligned}$$

- Because $|stu| \leq m$ this substring cannot contain both 0 and 2.
- Because $su \neq \varepsilon$ either s or u must contain at least one symbol from Σ.
- Thus rtv does not contain the same number of each symbol from Σ.
- This is a contradiction, because $rtv \in L$.

What is the smallest possible value of "m" for a *non-empty* context-free language L over Σ?

$$
\begin{aligned}
&\exists m \in \mathbb{N}. \\
&\quad \forall w \in L.\ |w| \geq m \Rightarrow \\
&\qquad \exists r, s, t, u, v \in \Sigma^*. \\
&\qquad\quad w = rstuv \wedge |stu| \leq m \wedge su \neq \varepsilon \wedge \\
&\qquad\quad \forall n \in \mathbb{N}.\ rs^n tu^n v \in L
\end{aligned}
$$

Today

- Grammar transformations.
- Chomsky normal form.
- The pumping lemma for context-free languages.

Next lecture

- Closure properties.
- Algorithms.