Week 10 - CFGs & Structural Induction

Billy Price

CONTEXT FREE GRAMMARS

Definition A context-free language (a CFL) is a language described by a CFG

CFGs are to Context Free Languages what regular expressions are to Regular Languages. They describe a recipe for creating strings of the language.

A context-free grammar (CFG) G is a 4-tuple (V, Σ, R, S), where

- V is a finite set of variables,
- Σ is a finite set of terminals,
- R is a finite set of rules, each consisting of a variable (the left-hand side) and a string in $(V \cup \Sigma)^*$ (the right-hand side),
- S is the start variable.

The binary relation $\Rightarrow$ on $(V \cup \Sigma)^*$ is defined as follows.

Let $u, v, w \in (V \cup \Sigma)^*$. Then $uAw \Rightarrow uvw$ iff $A \to v$ is a rule in R. That is, $\Rightarrow$ captures a single derivation step.

Let $\overset{*}{\Rightarrow}$ be the reflexive transitive closure of $\Rightarrow$.

$$L(G) = \{s \in \Sigma^* \mid S \overset{*}{\Rightarrow} s\}$$

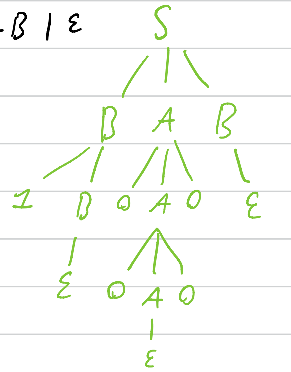
Example $G := (\{S, A, B\}, \{0, 1\}, R, S)$

$S \to A \mid BAB$

$A \to 0A0 \mid \varepsilon$

$B \to 1B \mid \varepsilon$

$10000 \in L(G)$



$S \Rightarrow BAB$
$\Rightarrow 1BAB$
$\Rightarrow 1\varepsilon AB$
$\Rightarrow 1A\varepsilon$
$\Rightarrow 10A0$
$\Rightarrow 100A00$
$S \overset{*}{\Rightarrow} 10000$

STRUCTURAL INDUCTION

Say we have some property $P(-)$ about strings over some alphabet, Σ, and a context-free language $A \subseteq \Sigma^*$. How do we show $P(w)$ for all $w \in A$?

Structural Induction! (on the grammar) Given a CFG, G, for A, look at the rules involving the starting variable S.

① Identify the base case rules and recursive case rules for the starting variable

② Prove $P(w)$ for each base case rule $S \to w$

③ For each of the recursive case rules, $S \to X$ replace all of the occurences of S in X with a string variable (u, v, x, y etc). Then, assuming $P(u), P(v), P(x), P(y) \ldots$ are all true, prove P is true of the whole string.

Example $S \to aaa \mid b \mid c \mid aSb \mid cSS$

Base case Rules	Recursive Case Rules
$S \to aaa$	$S \to aSb$
$S \to b$	$S \to cSS$
$S \to c$	

Step 2
- Prove $P(aaa)$
- Prove $P(b)$
- Prove $P(c)$

Step 3
- Assuming $P(u)$, prove $P(aub)$. $L(G)$
- Assuming $P(x)$ & $P(y)$, prove $P(cxy)$