Closure properties of CFL

✓ Let L_1 and L_2 be two CFL generated by CFG $G_1 = (V_1, T_1, S_1, P_1)$ and $G_2 = (V_2, T_2, S_2, P_2)$

$G = G_1 \cup G_2$	$G = G_1 \cdot G_2$	$G = G_1^*$
$S \rightarrow S_1 \mid S_2$	$S \rightarrow S_1 S_2$	$S \rightarrow S_1^*$

CFG are closed under union, kleene closure and concatenation

✓ CFL are not closed under intersection and complementation

$L_1 = \{a^n b^n c^m : n \geq 0, m \geq 0\}$

and

$L_2 = \{a^n b^m c^m : n \geq 0, m \geq 0\}$

Ex 1) $L = L_1 \cap L_2 = \{a^n b^n c^n : n \geq 0\}$

which is not CFG.

Ex.2

$L_1 = a^n b^n c^m d^p, \quad n, m, p \geq 0$

$L_2 = a^m b^k c^p d^p \quad m, k, p \geq 0$

$L = L_1 \cap L_2 = a^n b^n c^m d^m \quad n, m \geq 0$

which is CFL

Intersection

$$L_1 \cap L_2 = \overline{\overline{L_1} \cup \overline{L_2}}$$

if CFL are closed under complementation then the right side of above expression ~~who~~ would be CFL.

CFL are not closed under ~~to~~ intersection therefore CFL is not closed under complementation

CFL and Regular language

let L_1 be CFL and L_2 be regular language. Then $L_1 \cap L_2$ is CFL

Let $M_1 = (Q, \Sigma, \Gamma, \delta_1, q_0, Z, F_1)$ be NPDA which accept L_1 and

$M_2 = (P, \Sigma, \delta_2, P_0, F_2)$ be DFA that accept L_2

we construct PDA $\hat{M} = (\hat{Q}, \Sigma, \Gamma, \hat{\delta}, \hat{q_0}, Z, F)$ which simulates the parllel action of M_1 and M_2

$\hat{Q} = Q \times P$ $\quad F = F_1 \times F_2$

$\hat{q_0} = (q_0, P_0)$

$\hat{\delta}$ is defined as

$((q_K, P_l), x) \in \hat{\delta}((q_i, P_J), a, b)$ when

DATE: / /

$\delta_1(q_i, a, b) \vdash (q_k, x)$

and $\delta_2(P_j, a) \vdash (P_k)$

Therefore a string is accepted by M if and only if it is accepted by M_1 and M_2 that is if it is

$$L(M_1) \cap L(M_2) = L_1 \cap L_2$$

Ex. Show that

$$L = a^n b^n : n \geq 0, \; n \neq 100 \quad \text{is CFG.}$$

Proof

let $L_1 = a^{100} b^{100}$

L_1 is finite so it is regular

$$L = (a^n b^n \; n \geq 100) \cap \overline{L_1}$$

$\overline{L_1}$ is also regular

CFG $\cap$ regular = CFG

so L is CFG.

Decidable properties of context free language → a problem is decidable if there exist a solution to solve

Theorm 1. Given a CFG $G = (V, T, S, P)$ there exist an algorithm for deciding whether $L(G)$ is empty or not

Proof. Assume that $\Lambda \notin L(G)$
→ Find useless symbol if S (start) is found to be useless than $L(G)$ is empty

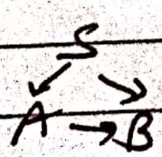
Theorm 2 $L(G)$ is infinite

Proof. Create a dependancy of Non-Terminal based on the production

→ if graph contain cycle than $L(G)$ is infinite else it is finite

Ex-1

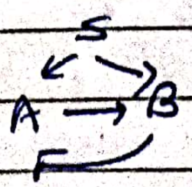
$S \to AB$
$A \to aB \mid b$
$B \to c \mid d$



No cycle so finite

Ex-2

$S \to AB$
$A \to aB \mid b$
$B \to bA \mid d$



cycle so infinite

Decidable properties for Regular language

1. L(G) is infinite

→ If there exist a cycle in the path from initial state to final state than L(G) is infinite

2. $L(G) = \phi$

L is empty if there exist no path from initial to final state

Scanned with CamScanner