

INF155: Informática Teórica

Clase 16: Símbolos Inútiles y Propiedades de Clausura para CFL

Aldo Berrios Valenzuela

Jueves 28 de Abril de 2016

1. Eliminar Símbolos Inútiles

Definición 1.1. Se dice que X es generante en la CFG $G = (N, \Sigma, P, S)$, con $X \in N \cup \Sigma$ si $X \Rightarrow^* \alpha \in \Sigma^*$

Definición 1.2. Se dice que X es alcanzable en CFG G si $S \Rightarrow^* \alpha X \beta$

Consideremos la CFG:

$$\begin{aligned} S &\rightarrow AB \mid a \\ A &\rightarrow b \end{aligned}$$

Todos salvo B son generantes, y todos son alcanzables.

Observación: Los terminales por derecho propio son generantes.

El orden en que eliminamos no generantes e inalcanzables importa:

- Primero eliminar no-generantes. Resultado:

$$\begin{aligned} S &\rightarrow a \\ A &\rightarrow b \end{aligned}$$

$\rightsquigarrow A, b$ no alcanzables. Eliminamos los inalcanzables:

$$S \rightarrow a$$

- Primero eliminar inalcanzables (no hay), luego no generantes terminamos con:

$$\begin{aligned} S &\rightarrow a \\ A &\rightarrow b \end{aligned}$$

Terminamos con basura y hasta aquí llegamos. Por ende el orden importa

Identificar símbolos generantes:

- $\Sigma \subseteq$ generantes.
- Si hay un (no-terminal) X tal que

$$X \rightarrow A_1 A_2 \cdots A_n$$

y todos los A_i con $1 \leq i \leq n$ son generantes, X es generante.

Iterar este... Después, la gramática resultante es $G' = (N', \Sigma, P', S)$ con:

- $N' = N \cap \text{generantes}$
- P' : producciones que no mencionan no generantes.

Caso especial: Generantes = \emptyset :

$$\rightsquigarrow G' = (\{S\}, \Sigma, \emptyset, S), \mathcal{L}(G) = \emptyset$$

Identificar alcanzables:

- S es alcanzable ($S \Rightarrow^* S$)
- Si X es alcanzable, y

$$X \rightarrow A_1 A_2 \cdots A_n,$$

entonces A_1, \dots, A_n son alcanzables.

Iterar lo anterior, y el resultado nos dará la gramática.

$$G'' = (N'', \Sigma'', P'', S)$$

Donde:

- $N'' = N' \cap \text{alcanzables}$
- $\Sigma'' = \Sigma \cap \text{alcanzables}$
- P'' : producciones de P' que sólo incluyen alcanzables.

Caso especial: $G = (\{S\}, \emptyset, \emptyset, S)$ genera \emptyset .

Algunas cosas útiles para demostrar:

- Podemos suponer que la gramática no tiene producciones unitarias.
- Podemos suponer que la gramática viene dada en forma normal de Chomsky.
- Podemos suponer que la gramática no contiene símbolos inútiles.

2. Propiedades de Clausura para CFL

Teorema 2.1. Las CFL (context free language) son cerrados respecto operaciones regulares (unión, concatenación y estrella Kleene)

Ojo: La estrella Kleene es complicada porque produce ϵ ...

Demostración. Sean $L = \mathcal{L}(G_1)$, $L_2 = \mathcal{L}(G_2)$, con $G_1 = (N, \Sigma, P, S)$, $G_2 = (N_2, \Sigma, P_2, S_2)$.

Unión – Supongamos que $N_1 \cap N_2 = \emptyset$. Entonces generan:

$$L_1 \cup L_2 : G_u = (N_1 \cup N_2 \cup \{S\}, \Sigma, P_u, S)$$

$$\text{con } P_u = \{S \rightarrow S_1, S \rightarrow S_2\} \cup P_1 \cup P_2.$$

Concatenación – Para este caso tenemos:

$$L_1 \cdot L_2 : G_c = (N_1 \cup N_2 \cup \{S\}, \Sigma, P_c, S)$$

$$\text{con } P_c = \{S \rightarrow S_1 S_2\} \cup P_1 \cup P_2$$

Estrella Kleene – Se tiene:

$$L^* : G_a = (N_1 \cup \{S\}, \Sigma, P_a, S) \\ \text{con } P_a = \{S \rightarrow SS_1, S \rightarrow \epsilon\} \cup P_1.$$

□

Teorema 2.2. Los lenguajes de contexto libre **no** son cerrados respecto al complemento.

Demostración. Sabemos que $L = \{a^n b^n c^n : n \geq 1\}$ no es CFL. Sin embargo, \bar{L} es CFL, ya que:

$$\bar{L} = \overline{\mathcal{L}(a^* b^* c^*)} \cup \{a^i b^j c^k : i \neq j \neq k\}$$

Los lenguajes regulares son cerrados respecto de su complemento, por lo tanto $\overline{\mathcal{L}(a^* b^* c^*)}$ es regular y en consecuencia, de contexto libre.

El lenguaje de $\{a^i b^j c^k : i \neq j \neq k\}$ es un lenguaje de contexto libre. Para poder construir su gramática debemos considerar el caso simple: construir una gramática para:

$$\{a^i b^j : i \neq j\} \quad (2.1)$$

El conjunto de producciones de (2.1) es:

$$\begin{aligned} S &\rightarrow aSb \mid aAb \mid abB \\ A &\rightarrow a \mid aA \\ B &\rightarrow b \mid bB \end{aligned}$$

De la misma forma, podemos construir $\{a^i b^j c^k : i \neq j \neq k\}$ a través del siguiente conjunto de producciones:

$$\left. \begin{aligned} S &\rightarrow aSbC \mid aAb \\ A &\rightarrow a \mid aA \\ C &\rightarrow c \mid cC \end{aligned} \right\} \text{Permutaciones de esto dan la segunda parte}$$

Como existe una gramática de contexto libre para $\{a^i b^j c^k : i \neq j \neq k\}$, entonces puede generar un lenguaje de contexto libre (valga la redundancia).

Los lenguajes de contexto libre son cerrados respecto a la unión, por ende, \bar{L} también es de contexto libre. □

Importante: Cuando decimos que un conjunto de lenguajes no es cerrado respecto a una operación x , no quiere decir que siempre al aplicar x sobre un lenguaje del conjunto vamos a tener un conjunto que no es de contexto libre... tenga cuidado.

Consideremos el hecho de los lenguajes regulares. Sabemos que el complemento de un lenguaje regular sigue siendo regular. Según la Jerarquía de Chomsky, se tiene que los lenguajes regulares también son de contexto libre, por lo tanto, el complemento de **ESE** caso particular CFL sí da de contexto libre.

Teorema 2.3. CFL son cerrados respecto a la intersección con lenguajes regulares.

Demostración. Similar a la intersección entre lenguajes regulares:

- $L_1 = \mathcal{L}(M_1)$, $M_1 = (Q_1, \Sigma, \Gamma, \delta_1, q_1, Z_1, F_1)$ un PDA.
- $L_2 = \mathcal{L}(M_2)$, $M_2 = (Q_2, \Sigma, \delta_2, q_2, F_2)$ un NFA.

La idea es construir un PDA M que siga la pista de M_1 y M_2 en componentes del estado

$$M = (Q_1 \times Q_2, \Sigma, \Gamma, \delta, (q_1, q_2), Z_1, F_1 \times F_2)$$

donde construimos

$$\delta((q', q''), X, A) = \{((p', p''), \gamma) : (p', \gamma) \in \delta_1(q', X, A) \wedge p'' \in \delta_2(q'', X)\}$$

Entonces [furioso agitar de brazos] $\mathcal{L}(M) = \mathcal{L}(M_1) \cap \mathcal{L}(M_2)$

□