Pontificia Universidad Católica de Chile
Escuela de Ingeniería
Departamento de Ciencia de la Computación

IIC2223 - Teoría de Autómatas y Lenguajes Formales

Ayudantía 7

Franco Bruña y Dante Pinto
28 de Octubre, 2021

Pregunta 1

Sea $\Sigma = \{0,1\}^*$. Demuestre que los siguientes lenguajes $L \subseteq \Sigma$ son libres de contexto.

- $L = \{w = 0^n 1 0^n 1 \mid n \geq 0\}$

  Para demostrar que este lenguaje es libre de contexto debemos encontrar una CFG que lo defina. Sea:

$$\begin{aligned} \mathcal{G}: \quad & S \to X1 \\ & X \to 0X0 \mid 1 \end{aligned}$$

$L \subseteq \mathcal{L}(\mathcal{G})$:

Sea $w \in L \Rightarrow w = 0^k 1 0^k 1,\ sk \geq 0$. Luego, sabemos que debe existir la derivación:

$$S \Rightarrow X1 \Rightarrow 0X01 \Rightarrow 0^2X0^21 \Rightarrow ... \Rightarrow 0^kX0^k1 \Rightarrow 0^k10^k1$$
$$\Rightarrow w \in \mathcal{L}(\mathcal{G})$$

$\mathcal{L}(\mathcal{G}) \subseteq L$:

Sea $w \in \mathcal{L}(\mathcal{G})$. Por la forma de la gramática, sabemos que existe una derivación de $k \geq 2$ pasos dada por:

$$S \Rightarrow X1 \Rightarrow 0X01 \Rightarrow 0^2X0^21 \Rightarrow ... \Rightarrow 0^kX0^k1 \Rightarrow 0^k10^k1$$
$$\Rightarrow w = 0^k10^k1,\ k \geq 0$$

.

$$\Rightarrow w \in L$$

$\therefore \mathcal{L}(\mathcal{G}) = L$

- $L = \{w \in \{0,1\}^* \mid |w|_0 = |w|_1 \}$, dónde $|w|_a$ representa el número de símbolos a en w.

  Para demostrar que este lenguaje es libre de contexto debemos encontrar una CFG que lo defina. Sea:

$$\mathcal{G}: \quad S \to 0S1 \mid 1S0 \mid SS \mid \varepsilon$$

  $\mathcal{L}(\mathcal{G}) \subseteq L$:

  Sea $w \in \mathcal{L}(\mathcal{G})$. Por la forma de la gramática, sabemos que en cada paso de la derivación, se agrega igual cantidad de unos y de ceros (2 para los primeros dos pasos y 0 para los últimos dos), por lo que sin importar cuál sea la derivación que existe para $w \in \mathcal{L}(\mathcal{G})$, se cumplirá que $|w|_0 = |w|_1$ y por tanto $w \in L$.

  $L \subseteq \mathcal{L}(\mathcal{G})$:

  Sea $w = a_1...a_n \in L$. Definimos $w_k = a_1...a_k$ para $k \leq n$ y $d(k,w) = |w_k|_1 - |w_k|_0$ como la diferencia entre la cantidad de unos y la cantidad de ceros de una palabra w desde el inicio de esta hasta la k-ésima posición. Es claro que:

$$w = a_1...a_n \in L \ \Rightarrow \ d(n,w) = 0$$

  Luego, podemos hacer un análisis recursivo para construir una derivación para la palabra, separando en dos casos:

  1. $a_1 \neq a_n$:
     Caso base: $|w| = 0$:
     $$\Rightarrow \ w = \varepsilon \ \wedge \ S \to \varepsilon$$
     Por tanto, existe una derivación para w.
     Caso general:
     $$\exists w' \mid w' \in L \ \wedge \ |w'| = |w| - 2 \ \wedge \ (w = 0 \cdot w' \cdot 1 \vee w = 1 \cdot w' \cdot 0)$$
     Como $w' \in L$, y S es la variable inicial, podemos tomar su derivación y a partir de ella crear una derivación para 2 agregando como primer paso $S \to 1S0$ o $S \to 0S1$ según corresponda.
  2. $a_1 = a_n$:
     Sabemos que $d(n, w = 0)$, por lo que asumiendo sin pérdida de generalidad que $a_1 = a_n = 0$, tendremos que $d(1,w) = -1 \ \wedge \ d(n-1,w) = 1$. Luego, como cada letra de nuestra palabra solamente puede sumar o restar 1 al valor de d, debe existir i tal que $1 < i < n$ y $d(i,w) = 0$ (de lo contrario sería imposible que $d(n-1,w) = 1$).
     Además, como $d(n,w) = 0$ y $d(i,w) = 0$, separando w como $w = a_1...a_n = w_i \cdot w_i'$, con $w_i = a_1...a_i$ y $w_i' = a_{i+1}...a_n$, tendremos que:
     $$\begin{gathered} d(n,w) = d(i,w_i) + d(n-i,w_i') \ \wedge \ d(i,w_i) = d(i,w) \\ \Rightarrow d(i,w_i) = 0 \ \wedge \ d(n-i,w_i') = 0 \\ \Rightarrow w_i \in L \ \wedge \ w_i' \in L \end{gathered}$$
     Finalmente, como w_i y w_i' están en L, sabemos que existirán derivaciones para ellos, por lo que podremos crear una derivación para w comenzando con $S \to SS$, siguiendo la derivación de w_i en la S de la izquierda y la de w_i' en la S de la derecha.

  $\Rightarrow$ existe una derivación para todo $w \in L$, es decir $w \in \mathcal{L}(\mathcal{G})$

  $\therefore \mathcal{L}(\mathcal{G}) = L$

Pregunta 2

Demuestre que todo lenguaje regular es también un lenguaje libre de contexto.
Sea L un lenguaje regular. Sabemos que existirá un DFA $\mathcal{A} = (Q, \Sigma, \delta, q_0, F)$ tal que $\mathcal{L}(\mathcal{A}) = L$. Definiendo ahora la CFG $\mathcal{G} = (Q, \Sigma, P, q_0)$, con P dado por:

$$\begin{aligned} P = &\{p \to aq \mid \delta(p,a) = q\} \\ \cup &\{p \to \varepsilon \quad \mid p \in F\} \end{aligned}$$

Para demostrar que L es libre de contexto, bastará demostrar que $\mathcal{L}(\mathcal{A}) = \mathcal{L}(\mathcal{G})$.

1. $\mathcal{L}(\mathcal{A}) \subseteq \mathcal{L}(\mathcal{G})$:
   Sea $w = a_1...a_n \in \mathcal{L}(\mathcal{A})$. Sabemos que existe una ejecución de $\mathcal{A}$ sobre w dada por:

   $$\rho:\ q_0 \xrightarrow{a_1} q_1 \xrightarrow{a_2} ... \xrightarrow{a_n} q_n \ \wedge\ q_n \in F$$

   Luego, si $q_{k-1} \xrightarrow{a_k} q_k$ está en la ejecución, sabemos que $\delta(q_{k-1}, a_k) = q_k$, por lo que tendremos la producción $q_{k-1} \to a_k q_k$ en $\mathcal{G}$. Además, como $q_n \in F$, sabemos que también debe existir $q_n \to \varepsilon$ en P, por lo que podemos generar la derivación:

   $$\begin{aligned} q_0 \Rightarrow a_1 q_1 \Rightarrow ... \Rightarrow a_1...a_n q_n \Rightarrow a_1...a_n \\ \Rightarrow w \in \mathcal{G} \end{aligned}$$

2. $\mathcal{L}(\mathcal{G}) \subseteq \mathcal{L}(\mathcal{A})$: Sea $w = a_1...a_n \in \mathcal{L}(\mathcal{G})$. Sabemos que existe una derivación para w dada por:

   $$q_0 \Rightarrow a_1 q_1 \Rightarrow ... \Rightarrow a_1...a_n q_n \Rightarrow a_1...a_n$$

   Sin embargo, por definición de $\mathcal{G}$, si $q_{k-1} \to a_k q_k$ está en P, se cumple que $\delta(q_{k-1}, a_k) = q_k$. Luego, tendremos la siguiente ejecución de $\mathcal{A}$ sobre w

   $$\rho:\ q_0 \xrightarrow{a_1} q_1 \xrightarrow{a_2} ... \xrightarrow{a_n} q_n \ \wedge\ q_n$$

   Finalmente, como $q_n \to \varepsilon$ está en P, se cumple que $q_n \in F$, por lo que la ejecución anterior es de aceptación y por tanto $w \in \mathcal{L}(\mathcal{A})$.

Por tanto, todo lenguaje regular es libre de contexto.

Pregunta 3

Demuestre que al eliminar las producciones unitarias y en vacío de una gramática libre de contexto, la cantidad de reglas resultantes es exponencial sobre la cantidad original.

Pregunta 4

Sea Σ un alfabeto. Para un lenguaje L sobre Σ se define $\texttt{SUFFIX}(L)$ como:

$$\texttt{SUFFIX}(L) = \{u \in \Sigma^* \mid \exists v \in \Sigma^*.\ vu \in L\}$$

Demuestre que si L es un lenguaje libre de contexto, entonces $\texttt{SUFFIX}(L)$ también es libre de contexto.
Sea L un lenguaje libre de contexto, luego existe $\mathcal{G} = (V, \Sigma, P, S)$ gramática libre de contexto en forma normal de Chomsky tal que $\mathcal{L}(\mathcal{G}) = L$. Definimos una nueva CFG $\mathcal{G}' = (V', \Sigma, P', S')$ tal que

- $V' = V \cup \{X_0 \mid X \in V\}$
- $P' = P \cup \{(X_0 \to Y_0 Z \mid Z_0) \mid (X \to YZ) \in P\} \cup \{(X_0 \to a \mid \epsilon) \mid (X \to a) \in P\}$
- $S' = S_0$

La idea de esta nueva CFG es que si a partir de una variable X se deriva una palabra w entonces a partir de X_0 se deriva w' tal que w' es sufijo de w. Para formalizar, introducimos el lenguaje

$$L_{\mathcal{G}}(X) = \{w \in \Sigma^* \mid X \Rightarrow^*_{\mathcal{G}} \ w\}$$

que contiene todas las palabras que se pueden derivar desde la variable X en la CFG $\mathcal{G}$. Es claro que para $\mathcal{G} = (V, \Sigma, P, S)$ se cumple $L_{\mathcal{G}}(S) = \mathcal{G}(\mathcal{L})$. Además, en el caso particular de $\mathcal{G}'$ tendremos que $L_{\mathcal{G}'}(S) = \mathcal{L}(\mathcal{G})$ por como definimos $\mathcal{G}'$ (en cierta forma, la gramática $\mathcal{G}$ está contenida dentro de $\mathcal{G}'$ y no agregamos producciones nuevas a partir de las variables originales V).

Mostremos que $L_{\mathcal{G}'}(X_0) = \texttt{SUFFIX}(L_{\mathcal{G}'}(X))$

- $\texttt{SUFFIX}(L_{\mathcal{G}'}(X)) \subseteq L_{\mathcal{G}'}(X_0)$ (Todo sufijo es generado por X_0):

  Sea $w' \in \texttt{SUFFIX}(L_{\mathcal{G}'}(X))$ el sufijo de una palabra w generada por la variable X. Usando inducción en el árbol de derivación de w (que tiene por raíz X) mostramos que w' es generado a partir de X_0, es decir, $w' \in L_{\mathcal{G}'}(X_0)$

  - Si $X \to a$, entonces $w = a$. En este caso $w' = \epsilon$ o $w' = a$, de manera que w' puede ser generado ya sea mediante $X_0 \to \epsilon$ o $X_0 \to a$.
  - Si $X \to YZ$, entonces $w = yz$ para $y, z \in \Sigma^*$ generados por Y y Z respectivamente. La hipótesis de inducción dice que cada sufijo de y es generado por Y_0 y que cada sufijo de z es generado por Z_0. Ahora, como w' es sufijo de w entonces o bien w' es sufijo de z o $w' = y'z$ para algún sufijo y' de y. Entonces, podemos generar w' usando o $X_0 \to Z_0$ o $X_0 \to Y_0 Z$

- $L_{\mathcal{G}'}(X_0) \subseteq \texttt{SUFFIX}(L_{\mathcal{G}'}(X))$ (Todo generado por X_0 es sufijo):

  Sea $w' \in L_{\mathcal{G}'}(X_0)$ palabra generada por la variable X_0. Usando inducción en el árbol de derivación de w' (que tiene por raíz X_0) demostraremos que w' debe ser sufijo de una palabra w generada por X, es decir, $w' \in \texttt{SUFFIX}(L_{\mathcal{G}'}(X))$.

  - Si $X_0 \to \epsilon$, entonces $w' = \epsilon$ que es sufijo de cualquier palabra. En particular como X es variable generadora entonces podemos elegir w como cualquier palabra generada por X.
  - Si $X_0 \to a$, entonces $w' = a$ y sabemos que $X \to a$ debe estar en P' por la construcción de este conjunto. Luego $w = a$ es generada por X.
  - Si $X_0 \to Z_0$, entonces $w' = z'$ para $z' \in \Sigma^*$ generada por Z_0. La hipótesis de inducción dice que z' es sufijo de una palabra z generada por Z. Además sabemos que $X \to YZ$ debe ser regla de la gramática. Como Y es variable generadora, entonces genera una palabra y. Así, tomamos $w = yz$ y es claro entonces que w' es sufijo de w y que w es generada por X usando la regla $X \to YZ$.
  - Si $X_0 \to Y_0 Z$ entonces $w' = y'z$ para y', z palabras generadas por Y_0 y Z, respectivamente. Nuevamente, por hipótesis de inducción tendremos que y' será sufijo de alguna palabra y generada por Y. Ahora elegimos $w = yz$ y nuevamente es fácil ver que w' es sufijo de w y que w es generada por X mediante $X \to YZ$.

De manera que efectivamente $L_{\mathcal{G}'}(X_0) = \texttt{SUFFIX}(L_{\mathcal{G}'}(X))$. Si usamos S entonces tendremos

$$\begin{aligned} L_{\mathcal{G}'}(S_0) &= \texttt{SUFFIX}(L_{\mathcal{G}'}(S)) \\ \mathcal{L}(\mathcal{G}') &= \texttt{SUFFIX}(\mathcal{L}(\mathcal{G})) \\ \mathcal{L}(\mathcal{G}') &= \texttt{SUFFIX}(L) \end{aligned}$$

Y por lo tanto, $\texttt{SUFFIX}(L)$ es lenguaje libre de contexto.
Nota: Solución redactada por Nicolás Van Sint Jan.