Introducción a la Lógica y la Computación

Mariana Badano Héctor Gramaglia
Pedro Sánchez Terraf Mauricio Tellechea
Guido Ivetta

FaMAF, 29 de octubre de 2021

Contenidos estimados para hoy

1. Repaso
   - Expresiones regulares
   - Operaciones con lenguajes
   - $Regex$ definen lenguajes regulares
2. Teorema de Kleene
   - Algoritmo

Operaciones con lenguajes

vacío	$\emptyset$	$\emptyset$
singulete	$\{x\} \quad (x \in \Sigma)$	x
complemento	$L^c = \Sigma^* \setminus L$	
intersección	$L \cap L' = \{\alpha \mid \alpha \in L \text{ y } \alpha \in L'\}$	
unión	$L \cup L' = \{\alpha \mid \alpha \in L \text{ ó } \alpha \in L'\}$	$r + r'$
concatenación	$L\,L' = \{\alpha\beta \mid \alpha \in L \text{ y } \beta \in L'\}$	$r\,r'$
potencias	$L^n = \begin{cases} \{\epsilon\} & \text{si } n = 0 \\ L\,L^k & \text{si } n = k+1 \end{cases}$	$r^0 = \epsilon$ r^n
clausura	$L^* = \bigcup_{n \in \mathbb{N}} L^n$	r^*

Expresiones regulares

Fijado un alfabeto Σ, son el menor conjunto $Regex$ que cumple:

$$
\begin{array}{rl}
\text{vacío} & \emptyset \in Regex. \\
\text{épsilon} & \boldsymbol{\epsilon} \in Regex. \\
\text{símbolo} & x \in \Sigma \implies \boldsymbol{x} \in Regex. \\
\text{unión} & r_1, r_2 \in Regex \implies r_1 + r_2 \in Regex. \\
\text{concatenación} & r_1, r_2 \in Regex \implies r_1\, r_2 \in Regex. \\
\text{clausura} & r \in Regex \implies r^* \in Regex.
\end{array}
$$

Expresiones regulares

Fijado un alfabeto Σ, son el menor conjunto $Regex$ que cumple:

vacío	$\boldsymbol{\emptyset} \in Regex$.
épsilon	$\boldsymbol{\epsilon} \in Regex$.
símbolo	$x \in \Sigma \implies \boldsymbol{x} \in Regex$.
unión	$r_1, r_2 \in Regex \implies r_1 + r_2 \in Regex$.
concatenación	$r_1, r_2 \in Regex \implies r_1\, r_2 \in Regex$.
clausura	$r \in Regex \implies r^* \in Regex$.

$$
\begin{aligned}
L(\boldsymbol{\emptyset}) &:= \emptyset \\
L(\boldsymbol{\epsilon}) &:= \{\epsilon\} \\
L(\boldsymbol{x}) &:= \{x\} \\
L(r_1 + r_2) &:= L(r_1) \cup L(r_2) \\
L(r_1\, r_2) &:= L(r_1)\, L(r_2) \\
L(r^*) &:= (L(r))^*
\end{aligned}
$$

Regex definen lenguajes regulares

Teorema

Para toda expresión regular r, existe un ϵ-NFA $\mathbb{A}(r)$ con un único estado final tal que $L(r) = L(\mathbb{A}(r))$.

Regex definen lenguajes regulares

Teorema

Para toda expresión regular r, existe un ϵ-NFA $\mathbb{A}(r)$ con un único estado final tal que $L(r) = L(\mathbb{A}(r))$.

Ahora, vemos la recíproca.

Teorema (Kleene)

Todo lenguaje regular es de la forma $L(r)$ para alguna $r \in Regex$.

Regex definen lenguajes regulares

Teorema

Para toda expresión regular r, existe un ϵ-NFA $\mathbb{A}(r)$ con un único estado final tal que $L(r) = L(\mathbb{A}(r))$.

Ahora, vemos la recíproca.

Teorema (Kleene)

Todo lenguaje regular es de la forma $L(r)$ para alguna $r \in Regex$.

Daremos un método recursivo para obtener a partir de un ϵ-NFA $\mathbb{A} = (\{q_0, \ldots, q_r\}, \Sigma, \delta, q_0, \{q_f\})$ con un único estado final, una expresión regular que denota el lenguaje $L(\mathbb{A})$.

Regex definen lenguajes regulares

Teorema

Para toda expresión regular r, existe un ϵ-NFA $\mathbb{A}(r)$ con un único estado final tal que $L(r) = L(\mathbb{A}(r))$.

Ahora, vemos la recíproca.

Teorema (Kleene)

Todo lenguaje regular es de la forma $L(r)$ para alguna $r \in Regex$.

Daremos un método recursivo para obtener a partir de un ϵ-NFA $\mathbb{A} = (\{q_0, \ldots, q_r\}, \Sigma, \delta, q_0, \{q_f\})$ con un único estado final, una expresión regular que denota el lenguaje $L(\mathbb{A})$.
La recursión será en $|Q| = |\{q_0, \ldots, q_r\}|$, así que iremos quitando estados para pasar a autómatas más chicos.

Algoritmo del Teorema de Kleene

- $\mathbb{A} = (\{q_0, \ldots, q_r\}, \Sigma, \delta, q_0, \{q_f\})$.
- $L(\mathbb{A}) =: L_{0f}(Q)$

UNC Universidad Nacional de Córdoba 1613 - 2013 400 AÑOS

Algoritmo del Teorema de Kleene

- $\mathbb{A} = (\{q_0, \ldots, q_r\}, \Sigma, \delta, q_0, \{q_f\})$.
- $L(\mathbb{A}) =: L_{0f}(Q) \rightsquigarrow$ palabras que arrancan en q_0 y terminan en q_f.

Algoritmo del Teorema de Kleene

- $\mathbb{A} = (\{q_0, \ldots, q_r\}, \Sigma, \delta, q_0, \{q_f\})$.
- $L(\mathbb{A}) =: L_{0f}(Q) \rightsquigarrow$ palabras que arrancan en q_0 y terminan en q_f.

En general, supongamos $R \subseteq Q$.

Definición

$L_{nm}(R) :=$ palabras que arrancan en q_n y terminan en q_m, involucrando sólo estados en R.

Algoritmo del Teorema de Kleene

- $\mathbb{A} = (\{q_0, \ldots, q_r\}, \Sigma, \delta, q_0, \{q_f\})$.
- $L(\mathbb{A}) =: L_{0f}(Q) \rightsquigarrow$ palabras que arrancan en q_0 y terminan en q_f.

En general, supongamos $R \subseteq Q$.

Definición

$L_{nm}(R) :=$ palabras que arrancan en q_n y terminan en q_m, involucrando sólo estados en R.

Pueden pasar por q_n varias veces

$\alpha = \alpha_1\alpha_2\alpha_3 \in L_{nm}(R) \quad \rightsquigarrow \quad q_n \xRightarrow{\alpha_1} q_n \xRightarrow{\alpha_2} q_n \xRightarrow{\alpha_3} q_m$

Algoritmo del Teorema de Kleene

- $\mathbb{A} = (\{q_0, \ldots, q_r\}, \Sigma, \delta, q_0, \{q_f\})$.
- $L(\mathbb{A}) =: L_{0f}(Q) \rightsquigarrow$ palabras que arrancan en q_0 y terminan en q_f.

En general, supongamos $R \subseteq Q$.

Definición

$L_{nm}(R) :=$ palabras que arrancan en q_n y terminan en q_m, involucrando sólo estados en R.

Pueden pasar por q_n varias veces

$\alpha = \alpha_1\alpha_2\alpha_3 \in L_{nm}(R) \rightsquigarrow q_n \xRightarrow{\alpha_1} q_n \xRightarrow{\alpha_2} q_n \xRightarrow{\alpha_3} q_m$

Definición

$I_n(R) :=$ palabras que salen y vuelven a q_n sin pasar en el medio por él (e involucrando sólo estados en R).

$F_{nm}(R) :=$ palabras que salen de q_n y llegan a q_m sin pasar nuevamente por q_n (e involucrando sólo estados en R).

Algoritmo del Teorema de Kleene, primera capa

- $L_{nm}(R) :=$ de q_n a q_m (sólo estados en R).
- $I_n(R) :=$ de q_n a sí mismo sin repetirlo (sólo estados en R).
- $F_{nm}(R) :=$ de q_n a q_m sin repetir q_n (sólo estados en R).

Algoritmo del Teorema de Kleene, primera capa

- $L_{nm}(R)$:= de q_n a q_m (sólo estados en R).
- $I_n(R)$:= de q_n a sí mismo sin repetirlo (sólo estados en R).
- $F_{nm}(R)$:= de q_n a q_m sin repetir q_n (sólo estados en R).

$$\overbrace{q_n \stackrel{\alpha_1}{\Longrightarrow}}^{I_n(R)} \overbrace{q_n \stackrel{\alpha_2}{\Longrightarrow}}^{I_n(R)} \overbrace{q_n \stackrel{\alpha_3}{\Longrightarrow} q_m}^{F_{nm}(R)}$$

Algoritmo del Teorema de Kleene, primera capa

- $L_{nm}(R)$:= de q_n a q_m (sólo estados en R).
- $I_n(R)$:= de q_n a sí mismo sin repetirlo (sólo estados en R).
- $F_{nm}(R)$:= de q_n a q_m sin repetir q_n (sólo estados en R).

$$\overbrace{q_n \overset{\alpha_1}{\Longrightarrow}}^{I_n(R)} \overbrace{q_n \overset{\alpha_2}{\Longrightarrow}}^{I_n(R)} \overbrace{q_n \overset{\alpha_3}{\Longrightarrow} q_m}^{F_{nm}(R)}$$

Luego

$$L_{nn}(R) := I_n(R)^*$$
$$L_{nm}(R) := I_n(R)^* F_{nm}(R) \qquad \text{si } n \neq m$$

Algoritmo del Teorema de Kleene, ciclos iniciales

- $L_{nm}(R) :=$ de q_n a q_m (sólo estados en R).
- $I_n(R) :=$ de q_n a sí mismo sin repetirlo (sólo estados en R).
- $F_{nm}(R) :=$ de q_n a q_m sin repetir q_n (sólo estados en R).

Algoritmo del Teorema de Kleene, ciclos iniciales

- $L_{nm}(R) :=$ de q_n a q_m (sólo estados en R).
- $I_n(R) :=$ de q_n a sí mismo sin repetirlo (sólo estados en R).
- $F_{nm}(R) :=$ de q_n a q_m sin repetir q_n (sólo estados en R).

$$q_n \xRightarrow[\cancel{q_n}]{a\,\beta\,b} q_n$$

Algoritmo del Teorema de Kleene, ciclos iniciales

- $L_{nm}(R) :=$ de q_n a q_m (sólo estados en R).
- $I_n(R) :=$ de q_n a sí mismo sin repetirlo (sólo estados en R).
- $F_{nm}(R) :=$ de q_n a q_m sin repetir q_n (sólo estados en R).

$$q_n \xRightarrow[\cancel{q_n}]{a\,\beta\,b} q_n \quad \rightsquigarrow \quad q_n \xrightarrow{a} q_t \xRightarrow{\beta} q_s \xrightarrow{b} q_n$$

Algoritmo del Teorema de Kleene, ciclos iniciales

- $L_{nm}(R) :=$ de q_n a q_m (sólo estados en R).
- $I_n(R) :=$ de q_n a sí mismo sin repetirlo (sólo estados en R).
- $F_{nm}(R) :=$ de q_n a q_m sin repetir q_n (sólo estados en R).

$$q_n \underset{\cancel{q_n}}{\overset{a\,\beta\,b}{\Longrightarrow}} q_n \quad \rightsquigarrow \quad q_n \xrightarrow{a} \underbrace{q_t \overset{\beta}{\Longrightarrow} q_s}_{L_{ts}(R\setminus\{q_n\})} \xrightarrow{b} q_n$$

Algoritmo del Teorema de Kleene, ciclos iniciales

- $L_{nm}(R)$:= de q_n a q_m (sólo estados en R).
- $I_n(R)$:= de q_n a sí mismo sin repetirlo (sólo estados en R).
- $F_{nm}(R)$:= de q_n a q_m sin repetir q_n (sólo estados en R).

$$q_n \underset{\cancel{q_n}}{\overset{a\,\beta\,b}{\Longrightarrow}} q_n \quad \rightsquigarrow \quad q_n \xrightarrow{a} \underbrace{q_t \overset{\beta}{\Longrightarrow} q_s}_{L_{ts}(R \setminus \{q_n\})} \xrightarrow{b} q_n$$

Luego

$$I_n(R) := \sum_{\substack{q_n \xrightarrow{a} q_t \\ q_s \xrightarrow{b} q_n}} a\, L_{ts}(R \setminus \{q_n\})\, b$$

Algoritmo del Teorema de Kleene, ciclos iniciales

- $L_{nm}(R)$:= de q_n a q_m (sólo estados en R).
- $I_n(R)$:= de q_n a sí mismo sin repetirlo (sólo estados en R).
- $F_{nm}(R)$:= de q_n a q_m sin repetir q_n (sólo estados en R).

$$q_n \underset{\cancel{q_n}}{\overset{a\,\beta\,b}{\Longrightarrow}} q_n \quad \rightsquigarrow \quad q_n \xrightarrow{a} \underbrace{q_t \overset{\beta}{\Longrightarrow} q_s}_{L_{ts}(R \setminus \{q_n\})} \xrightarrow{b} q_n$$

Luego

$$I_n(R) := \sum_{\substack{q_n \xrightarrow{a} q_t \\ q_s \xrightarrow{b} q_n}} a\, L_{ts}(R \setminus \{q_n\})\, b \; + \sum_{q_n \xrightarrow{c} q_n} c \qquad (n \neq t, s)$$

Algoritmo del Teorema de Kleene, camino al final

- $L_{nm}(R) :=$ de q_n a q_m (sólo estados en R).
- $I_n(R) :=$ de q_n a sí mismo sin repetirlo (sólo estados en R).
- $F_{nm}(R) :=$ de q_n a q_m sin repetir q_n (sólo estados en R).

Algoritmo del Teorema de Kleene, camino al final

- $L_{nm}(R) :=$ de q_n a q_m (sólo estados en R).
- $I_n(R) :=$ de q_n a sí mismo sin repetirlo (sólo estados en R).
- $F_{nm}(R) :=$ de q_n a q_m sin repetir q_n (sólo estados en R).

$$q_n \overset{a\,\beta}{\underset{\cancel{q_n}}{\Longrightarrow}} q_m$$

Algoritmo del Teorema de Kleene, camino al final

- $L_{nm}(R) :=$ de q_n a q_m (sólo estados en R).
- $I_n(R) :=$ de q_n a sí mismo sin repetirlo (sólo estados en R).
- $F_{nm}(R) :=$ de q_n a q_m sin repetir q_n (sólo estados en R).

$$q_n \underset{\cancel{q_n}}{\overset{a\,\beta}{\Longrightarrow}} q_m \quad \rightsquigarrow \quad q_n \xrightarrow{a} q_t \overset{\beta}{\Longrightarrow} q_m$$

Algoritmo del Teorema de Kleene, camino al final

- $L_{nm}(R) :=$ de q_n a q_m (sólo estados en R).
- $I_n(R) :=$ de q_n a sí mismo sin repetirlo (sólo estados en R).
- $F_{nm}(R) :=$ de q_n a q_m sin repetir q_n (sólo estados en R).

$$q_n \overset{a\,\beta}{\underset{\cancel{q_n}}{\Longrightarrow}} q_m \quad \rightsquigarrow \quad q_n \xrightarrow{a} \underbrace{q_t \overset{\beta}{\Longrightarrow} q_m}_{L_{tm}(R \setminus \{q_n\})}$$

Algoritmo del Teorema de Kleene, camino al final

- $L_{nm}(R)$:= de q_n a q_m (sólo estados en R).
- $I_n(R)$:= de q_n a sí mismo sin repetirlo (sólo estados en R).
- $F_{nm}(R)$:= de q_n a q_m sin repetir q_n (sólo estados en R).

$$q_n \xRightarrow[\cancel{q_n}]{a\,\beta} q_m \quad \rightsquigarrow \quad q_n \xrightarrow{a} \underbrace{q_t \xRightarrow{\beta} q_m}_{L_{tm}(R \setminus \{q_n\})}$$

De igual modo,

$$F_{nm}(R) := \sum_{q_n \xrightarrow{a} q_t} a\, L_{tm}(R \setminus \{q_n\}) \qquad (n \neq t, m)$$

No olvidar los casos base

Algoritmo recursivo de Kleene

$$L_{nm}(R) := \emptyset \qquad \text{si } q_n \text{ ó } q_m \text{ no están en } R$$

$$L_{nn}(R) := I_n(R)^*$$

$$L_{nm}(R) := I_n(R)^* F_{nm}(R) \qquad \text{si } n \neq m$$

$$I_n(R) := \sum_{\substack{q_n \xrightarrow{a} q_t \\ q_s \xrightarrow{b} q_n}} a\, L_{ts}(R \setminus \{q_n\})\, b \;+\; \sum_{q_n \xrightarrow{c} q_n} c \qquad (n \neq t, s)$$

$$F_{nm}(R) := \sum_{q_n \xrightarrow{a} q_t} a\, L_{tm}(R \setminus \{q_n\}) \qquad (n \neq t, m)$$

$$L_{nm}(R) := \emptyset \quad \text{si } \{q_n, q_m\} \not\subseteq R$$
$$L_{nn}(R) := I_n(R)^*$$
$$L_{nm}(R) := I_n(R)^* F_{nm}(R) \quad \text{si } n \neq m$$

$$I_n(R) := \sum_{\substack{q_n \xrightarrow{a} q_t \\ q_s \xrightarrow{b} q_n}} a\, L_{ts}(R \setminus \{q_n\})\, b \;+ \sum_{q_n \xrightarrow{c} q_n} c$$

$$F_{nm}(R) := \sum_{q_n \xrightarrow{a} q_t} a\, L_{tm}(R \setminus \{q_n\}) \qquad (n \neq t, s, m)$$