Parte III: Lenguajes y Autómatas

6 de noviembre de 2019

Hablemos de lenguajes

Puesto que un lenguaje es un subconjunto de Σ^* podemos hablar de:

vacío $\emptyset$

total Σ^*

complemento $\bar{L} = \Sigma^* \setminus L$

intersección $L \cap L' = \{\alpha \mid \alpha \in L \text{ y } \alpha \in L'\}$

unión $L \cup L' = \{\alpha \mid \alpha \in L \text{ o } \alpha \in L'\}$

concatenación $LL' = \{\alpha\beta \mid \alpha \in L \text{ y } \beta \in L'\}$

potencias $L^n = \begin{cases} \{\epsilon\} & \text{si } n = 0 \\ LL^k & \text{si } n = k+1 \end{cases}$

clausura $L^* = \bigcup_{n \in \mathbb{N}} L^n$

Expresiones regulares

Las expresiones regulares son una forma más algebraica de definir lenguajes regulares. Fijado un alfabeto Σ las definimos inductivamente:

vacío	$\emptyset \in ER_\Sigma$.
épsilon	$\epsilon \in ER_\Sigma$.
símbolo	Si $x \in \Sigma$, entonces $\mathbf{x} \in ER_\Sigma$.
unión	Si $A, B \in ER_\Sigma$, entonces $A + B \in ER_\Sigma$.
concatenación	Si $A, B \in ER_\Sigma$, entonces $AB \in ER_\Sigma$.
clausura	Si $A \in ER_\Sigma$, entonces $A^* \in ER_\Sigma$.

Lenguaje de una expresión regular

Como ER_Σ fue definido inductivamente, podemos definir sus lenguajes por recursión:

vacío $L(\emptyset) = \emptyset.$

épsilon $L(\epsilon) = \{\epsilon\}.$

símbolo $L(\mathbf{x}) = \{x\}.$

unión $L(A + B) = L(A) \cup L(B).$

concatenación $L(AB) = L(A)L(B).$

clausura $L(A^*) = (L(A))^*.$

Álgebra de ERs

Igualdades

$$\emptyset + A = A \qquad \text{(neutro de } +\text{)}$$

$$\epsilon A = A = A\epsilon \qquad \text{(neutro de .)}$$

$$\emptyset A = \emptyset = A\emptyset \qquad \text{(nulo de .)}$$

$$\emptyset^* = \epsilon \qquad \text{por neutro de .}$$

$$A + B = B + A \qquad \text{(conmutatividad)}$$

$$A + (B + C) = (A + B) + C \qquad \text{(asociatividad)}$$

$$A(B + C) = AB + AC \qquad \text{distributividad izq.}$$

$$(A + B)C = AC + BC \qquad \text{distributividad der.}$$

Teorema

Para toda expresión regular A, existe un AFN-ϵ M tal que $L(A) = L(M)$.

Demostración

Construimos (recursivamente) un AFN-ϵ con un *único* estado final.

Teorema (Kleene)

Para todo AFD M, existe una ER A tal que $L(M) = L(A)$.

Idea de demostración

Al consumir una palabra el autómata sale del estado inicial eventualmente vuelve al mismo, hasta que va hacia un estado final sin volver a pasar por el inicial. Damos ER para los ciclos y para otros caminos eliminando estados.

Eliminación de estados

En general, un camino del estado n al estado m se puede dividir en ciclos sobre n y luego salir de n para llegar a m sin pasar de nuevo por n.

$$L_{n\,m}(R) = (I_n(R))^* F_{n\,m}(R)$$

Un ciclo sobre n lo dividimos en ciclos triviales y en las formas de salir a s y llegar a n desde otro estado t, sin volver a pasar por n en el medio:

$$I_n(R) = \sum_{n \xrightarrow{c} n} c + \sum_{n \xrightarrow{a} s,\, t \xrightarrow{b} n} a L_{s\,t}(R \setminus \{n\}) b$$

Un camino de n a m sin pasar por n son caminos que salen de n a un estado s y de allí va a m sin pasar por n:

$$F_{n\,m}(R) = \sum_{n \xrightarrow{a} s} a L_{s\,m}(R \setminus \{n\})$$

Eliminación de estados

En esta filmina f, m, n, s, t son estados, R es un conjunto de estados.

$$L_{n\,m}(R) = \emptyset \quad \text{si } n \notin R \text{ ó } m \notin R$$

$$L_{n\,m}(R) = \begin{cases} (I_n(R))^* & \text{si } n = m \\ (I_n(R))^* F_{n\,m}(R) & \text{si } n \neq m \end{cases}$$

$$I_n(R) = \sum_{n \xrightarrow{c} n} c + \sum_{n \xrightarrow{a} s, t \xrightarrow{b} n} a L_{s\,t}(R \setminus \{n\}) b$$

$$F_{n\,m}(R) = \sum_{n \xrightarrow{a} s} a L_{s\,m}(R \setminus \{n\})$$

La ER completa suma las ER desde el estado inicial a los estados finales

$$L(M) = \sum_{f \in F} L_{0\,f}(Q)$$