Sesión 6

El lenguaje aceptado por un FA

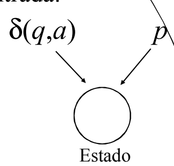
Función de transición δ

z	p_j	…	p_l
…	…	…	…
a	p_i	…	p_k
Σ \ Q	q_0	…	q_n

Para todo q en Q & $a \in \Sigma$, $\delta(q, a) = p$

Función de transición extendida

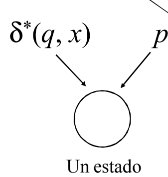
- δ permite moverse the un estado a otro con un símbolo de entrada
- δ nos da un medio para referirse o nombrar al estado siguiente en términos del estado actual y el símbolo de entrada:
  - $\delta(q,a) = p$
- Relación de referencia:



Función de transición extendida

- Es muy conveniente contar con una forma de referirse al siguiente estado en función del estado actual y *una cadena de entrada*:
  - $\delta^*(q, x) = p$
- El estado p al que se llega a partir del estado actual q y una cadena dada
- La función de transición aumentada δ^* nos provee de este medio de referencia!

Función de transición extendida



Función de transición extendida δ^*

...	...	…	...
x_1	p_{01}	…	p_{n1}
$x_0 = \Lambda$	p_{00}	…	p_{n0}
Σ^* \ Q	q_0	…	q_n

Para todo q de Q & $x \in \Sigma^*$, $\delta^*(q, x) = p$

Definición formal de δ^*

- Sea $M = (Q, \Sigma, q_0, A, \delta)$ un FA
- Definimos la función $\delta^*: Q \times \Sigma^* \to Q$ como sigue:
  - Para todo $q \in Q$, $\delta^*(q, \Lambda) = q$
  - Para todo $q \in Q$ & cualquier $y \in \Sigma^*$ & un símbolo $a \in \Sigma$

$$\delta^*(q, ya) = \delta(\delta^*(q, y), a)$$

Operación de δ^*

$$q_0 \xrightarrow{a} q_1 \xrightarrow{b} q_2 \xrightarrow{c} q_3$$

- $\delta^*(q_0, abc) = q_3$

Operación de δ^*

$$q_0 \xrightarrow{a} q_1 \xrightarrow{b} q_2 \xrightarrow{c} q_3$$

- $\delta^*(q_0, abc) = \delta(\delta^*(q_0, ab), c)$

Operación de δ^*

$$q_0 \xrightarrow{a} q_1 \xrightarrow{b} q_2 \xrightarrow{c} q_3$$

- $\delta^*(q_0, abc) = \delta(\delta^*(q_0, ab), c)$
  $= \delta(\delta(\delta^*(q_0, a), b), c)$

Operación de δ^*

$$q_0 \xrightarrow{a} q_1 \xrightarrow{b} q_2 \xrightarrow{c} q_3$$

- $\delta^*(q_0, abc) = \delta(\delta^*(q_0, ab), c)$
  $= \delta(\delta(\delta^*(q_0, a), b), c)$
- Pero δ^* no está definida para cadenas de un solo caractér:

$$\delta^*(q, ya) = \delta(\delta^*(q, y), a)$$

Operación de δ^*

$$q_0 \xrightarrow{a} q_1 \xrightarrow{b} q_2 \xrightarrow{c} q_3$$

- $\delta^*(q_0, abc) = \delta(\delta^*(q_0, ab), c)$
  $= \delta(\delta(\delta^*(q_0, a), b), c)$
  $= \delta(\delta(\delta^*(q_0, \Lambda a), b), c)$
- Pero … podemos usar la identidad de la concatenación!

Operación de δ^*

$$\rightarrow q_0 \xrightarrow{a} q_1 \xrightarrow{b} q_2 \xrightarrow{c} q_3$$

- $\delta^*(q_0, abc) = \delta(\delta^*(q_0, ab), c)$
- $= \delta(\delta(\delta^*(q_0, a), b), c)$
- $= \delta(\delta(\delta^*(q_0, \Lambda a), b), c)$
- $= \delta(\delta(\delta(\delta^*(q_0, \Lambda), a), b), c)$

- Usando la defínción de δ^*:

  $\delta^*(q, ya) = \delta(\delta^*(q, y), a)$

Dr. Luis Pineda, IIMAS, UNAM, 2010

Operación de δ^*

$$\rightarrow q_0 \xrightarrow{a} q_1 \xrightarrow{b} q_2 \xrightarrow{c} q_3$$

- $\delta^*(q_0, abc) = \delta(\delta^*(q_0, ab), c)$
- $= \delta(\delta(\delta^*(q_0, a), b), c)$
- $= \delta(\delta(\delta^*(q_0, \Lambda a), b), c)$
- $= \delta(\delta(\delta(\delta^*(q_0, \Lambda), a), b), c)$

- Nada inusual: aplicamos la identidad de la concatenación

Dr. Luis Pineda, IIMAS, UNAM, 2010

Operación de δ^*

$$\rightarrow q_0 \xrightarrow{a} q_1 \xrightarrow{b} q_2 \xrightarrow{c} q_3$$

- $\delta^*(q_0, abc) = \delta(\delta^*(q_0, ab), c)$
- $= \delta(\delta(\delta^*(q_0, a), b), c)$
- $= \delta(\delta(\delta^*(q_0, \Lambda a), b), c)$
- $= \delta(\delta(\delta(\delta^*(q_0, \Lambda), a), b), c)$
- $= \delta(\delta(\delta(q_0, a), b), c)$
- Usando la condición básica: $\delta^*(q, \Lambda) = q$
- Con esto hemos reducido δ^* a una composición de δs

Dr. Luis Pineda, IIMAS, UNAM, 2010

Operación de δ^*

$$\rightarrow q_0 \xrightarrow{a} q_1 \xrightarrow{b} q_2 \xrightarrow{c} q_3$$

- $\delta^*(q_0, abc) = \delta(\delta^*(q_0, ab), c)$
- $= \delta(\delta(\delta^*(q_0, a), b), c)$
- $= \delta(\delta(\delta^*(q_0, \Lambda a), b), c)$
- $= \delta(\delta(\delta(\delta^*(q_0, \Lambda), a), b), c)$
- $= \delta(\delta(\delta(q_0, a), b), c)$
- $= \delta(\delta(q_1, b), c)$

Dr. Luis Pineda, IIMAS, UNAM, 2010

Operación de δ^*

$$\rightarrow q_0 \xrightarrow{a} q_1 \xrightarrow{b} q_2 \xrightarrow{c} q_3$$

- $\delta^*(q_0, abc) = \delta(\delta^*(q_0, ab), c)$
- $= \delta(\delta(\delta^*(q_0, a), b), c)$
- $= \delta(\delta(\delta^*(q_0, \Lambda a), b), c)$
- $= \delta(\delta(\delta(\delta^*(q_0, \Lambda), a), b), c)$
- $= \delta(\delta(\delta(q_0, a), b), c)$
- $= \delta(\delta(q_1, b), c)$
- $= \delta(q_2, c)$

Dr. Luis Pineda, IIMAS, UNAM, 2010

Operación de δ^*

$$\rightarrow q_0 \xrightarrow{a} q_1 \xrightarrow{b} q_2 \xrightarrow{c} q_3$$

- $\delta^*(q_0, abc) = \delta(\delta^*(q_0, ab), c)$
- $= \delta(\delta(\delta^*(q_0, a), b), c)$
- $= \delta(\delta(\delta^*(q_0, \Lambda a), b), c)$
- $= \delta(\delta(\delta(\delta^*(q_0, \Lambda), a), b), c)$
- $= \delta(\delta(\delta(q_0, a), b), c)$
- $= \delta(\delta(q_1, b), c)$
- $= \delta(q_2, c)$
- $= q_3$

Dr. Luis Pineda, IIMAS, UNAM, 2010

Como consecuencia de la definición de δ^*

- Composición de transición de cadenas:

$$\delta^*(q, xy) = \delta^*(\delta^*(q, x), y)$$

El lenguaje aceptado por un FA

- Sea $M = (Q, \Sigma, q_0, A, \delta)$ un FA
- $x \in \Sigma^*$ es aceptado por M Si $\delta^*(q_0, x) \in A$
- Si una cadena no es aceptada por M, se rechaza!
- El lenguaje *aceptado* o *reconocido* por M es el conjunto:
  - $\mathbf{L}(M) = \{x \in \Sigma^* \mid x$ es aceptada por $M\}$
- Si L es cualquier lenguage sobre Σ, L es aceptado, o reconocido, por M si y solo si $L = \mathbf{L}(M)$

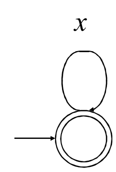
Dr. Luis Pineda, IIMAS, UNAM, 2010

Pero …

- Está definición no dice que L es aceptado por M *si* todas las cadenas de L son aceptadas por M
- Si fuera así, M podría aceptar también cadenas que no están en L!
- Un FA acepta un lenguaje si
  - Acepta todas las cadenas en L
  - Rechaza todas las cadenas en su complemento

Dr. Luis Pineda, IIMAS, UNAM, 2010

Pero …



Dr. Luis Pineda, IIMAS, UNAM, 2010

TEOREMA de Kleen

Un lenguaje L sobre el alfabeto Σ es regular si y sólo si existe un FA con alfabeto de entrada Σ que acepta a L

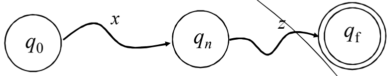
Dr. Luis Pineda, IIMAS, UNAM, 2010

Relación entre FA y *RE*

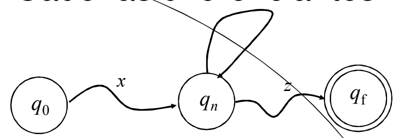
- Si esto es verdad (y no les quepa la menor duda), las expresiones regulares y los autómatas de estados finitos son representaciones equivalentes: Si una *ER denota* a un lenguaje, existe un FA que acepta dicho lenguaje!
  - *ER* son representaciones declarativas
  - FA son representaciones procedurales

Dr. Luis Pineda, IIMAS, UNAM, 2010

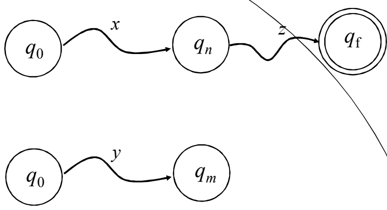
Cadenas diferenciantes



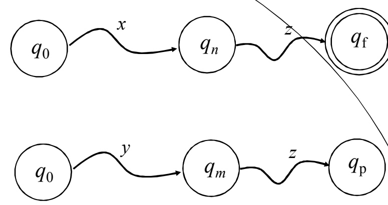
Cadenas diferenciantes



Cadenas diferenciantes



Cadenas diferenciantes

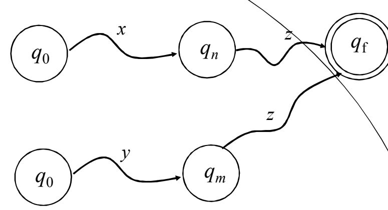


La cadena z diferencia a todas las cadenas x de las cadenas y en relación al lenguaje

Cadenas diferenciantes

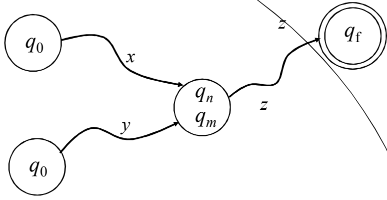
- Ejemplo: $L = \{w \mid |w| \text{ es par}\}$
  - Cualquier cadena z de longitud par distingue las cadenas de longitud par en L, de las cadenas de longitud non, que no están en L
  - L parte al universo de las cadenas en dos clases equivalentes:
    - Las cadenas de longitud par
    - Las cadenas de longitud non

Cadenas diferenciantes



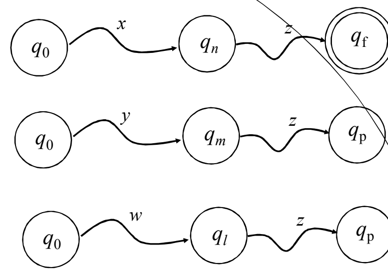
Si z no diferencia a x de y...

Cadenas diferenciantes



Todas las x y todos las y
son equivalentes en relación al lenguaje

Cadenas diferenciantes



Cadenas diferenciantes

- Sea L un lenguaje en Σ & $x \in \Sigma^*$
- Sea L/x un conjunto de cadenas tal que:
  - $L/x = \{z \in \Sigma^* \mid xz \in L\}$
- Dos cadenas x & y son diferenciables con respecto a L si
  - $L/x \neq L/y$
- Toda cadena z tal que $xz \in L$ & $yz \notin L$ o viceversa, distingue (diferencía) a x & y con respecto a L
- Si $L/x = L/y$, x & y son indistinguibles (equivalentes) con respecto a L

Cadenas diferenciantes

- Para mostrar que x and y son diferenciantes con respecto a L
  - Encontrar z tal que $xz \in L$ PERO $yz \notin L$ o
  - Encontrar z tal que $yz \in L$ PERO $xz \notin L$
  - Entonces z está en L/x o L/y pero no en ambos!

Cadenas diferenciantes

- Ejemplo:
  - $L = \{w \mid |w| \text{ es par}\}$
  - Cualquier cadena z de longitud par distingue las cadenas de longitud par en L, de las cadenas de longitud non, que no están en L
  - $L/x = \{z \in \Sigma^* \mid xz \in L\}$
  - Si $|x|$ es non $|xz|$ también es non ($|z|$ es par) & $xz \notin L$

Cadenas diferenciantes

- Lema:
  - Sea $L \subseteq \Sigma^*$ & $M = (Q, \Sigma, q_0, A, \delta)$ es un FA que reconoce L.
  - Si x & y son cadenas en Σ^* diferenciables con respecto a L (en relación a la cadena diferenciante z), entonces:

$$\delta^*(q_0, xz) \neq \delta^*(q_0, yz)$$

  - ¡Un estado es aceptor pero no el otro!

Prueba del lema

- Un estado es aceptor pero no el otro:
  - $\delta^*(q_0, xz) \neq \delta^*(q_0, yz)$
- Los estados alcanzados a partir de q_0 con xz & con yz son, respectivamente:
  - $\delta^*(q_0, xz) = \delta^*(\delta^*(q_0, x), z)$
  - $\delta^*(q_0, yz) = \delta^*(\delta^*(q_0, y), z)$

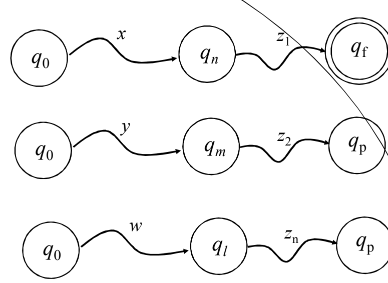
Prueba del lema

- Ahora bien, si z es una cadena diferenciante:
  - $\delta^*(\delta^*(q_0, x), z) \neq \delta^*(\delta^*(q_0, y), z)$
- Entonces:
  - $\delta^*(q_0, x) \neq \delta^*(q_0, y)$
- Esto es: x & y llevan a M desde q_0 a estados diferentes!

Teorema

- Supongamos que $L \subseteq \Sigma^*$ y para cierta n, hay n cadenas en Σ^*, tales que cualquier par formado por estas cadena es distinguible (diferenciable) con respecto a L.
- Entonces, todo FA que reconozca a L debe tener cuando menos n estados.

Cadenas diferenciantes



Prueba del teorema

- Supongamos que $x_1, x_2, \ldots, x_n$ son n cadenas, tales que cualquier par formado por éstas es distinguible con respecto a L
- Entonces, si $M = (Q, \Sigma, q_0, A, \delta)$ es un FA con menos de n estados NO todos los estados
  - $\delta^*(q_0, x_1)$
  - $\delta^*(q_0, x_2)$
  - ….
  - $\delta^*(q_0, x_n)$

  son diferentes!

Prueba del teorema

- Ya que para algún $i \neq j$,

  $\delta^*(q_0, x_i) = \delta^*(q_0, x_j)$
- Pero, dado que x_i & x_j son diferenciables con respecto a L, se sigue del lema que M no puede reconocer a L

¿Cuando un lenguaje no es regular?

- El teorema de las cadenas diferenciantes nos permite determinar cuando un lenguaje no es regular
- Si existe un número infinito de cadenas tales que cualquier par formado por las mismas es distinguible con respecto a *L*, cualquier FA que acepte *L* debe tener un número infinito de estados...
- Pero la "F" de "FA" quiere decir FINITOS
- Por lo tanto, ningún FA puede aceptar un lenguaje con un número infinito de cadenas diferenciantes!

Un lenguaje no aceptado por un FA

- El lenguaje *pal* sobre $\Sigma = \{0, 1\}$ no se acepta por ningún FA y por lo tanto, no es regular
- Palíndrome: *eva ama ave*
  - Palíndromes: Λ, 0, 1, 0110, 11011
- Prueba: Mostramos que cualquier par de cadenas x & y en $\{0, 1\}^*$ son diferenciables con respecto a *pal*

Un lenguaje no aceptado por un FA

- Caso 1: $|x| = |y|$ & sea $z = x^r$
  - Entonces $xz = xx^r$ es una palíndrome
  - Si $x = 0101$ & $y = 1111$

    $z = x^r = 1010$ & $xz = 01011010$ (*pal*)
  - Pero yz no lo es:

    $yz = 11111010$
  - $z = x^r$ distingue a x de y para cualquier y
  - Cualquier cadena puede distinguirse de todas las demás de su misma longitud con respecto a *pal*

Un lenguaje no aceptado por un FA

- Caso 2: $|x| \neq |y|$ & asumimos $|x| < |y|$
  - Sea $y = y_1y_2$ donde $|y_1| = |x|$
  - Buscamos una z tal que:

    $xz \in pal$ pero $yz \notin pal$
  - Cualquier z de forma:

    $z = ww^rx^r$

    satisface $xz \in pal$
  - Esto es: $xz = xww^rx^r$ (i.e. $x\mathrm{P}x^r$)

Un lenguaje no aceptado por un FA

- Seleccionamos w (asegurandonos que $yz \notin pal$):
  - $yww^rx^r \notin pal$ o $y_1y_2ww^rx^r \notin pal$
  - Consideramos que $|y_1| = |x|$ por lo que y_1 podría ser x (e.g. $xy_2ww^rx^r$)
  - Escogemos una w tal que $|w| = |y_2|$ pero $y_2ww^r \notin pal$
  - Por lo mismo basta con que $w \neq y_2$ (en caso de que accidentalmente $w \in pal$ ya que www^r sería una palindroma)

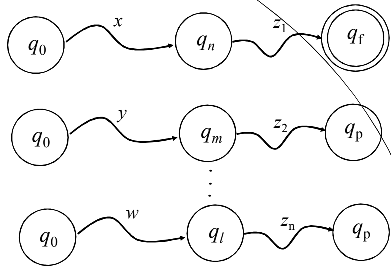
Un lenguaje que no es aceptado por un FA

- Caso 2: $|x| \neq |y|$ & asumimos $|x| < |y|$
  - (e.g. $x = 010$ & $y = 10101$)
  - Escogemos $w = 10$ (i.e. $w \neq y_2$)
  - Consecuentemente $xz = xww^rx^r$

    (i.e. $xwwx^r = 0101001010$)
  - Pero $yz = y_1y_2z = y_1y_2ww^rx^r$

    (i.e. 101011001010)

Cada cadena se puede diferenciar con respecto a *pal*



Cadenas diferenciadoras

- ¿Cuántas clases diferentes hay en un lenguaje?
- Para $L = \{w \mid |w| \text{ es par}\}$ hay dos clases de cadenas: pares y nones
- Una sóla cadena (par) basta para distinguir a las dos clases

Cadenas diferenciadoras

- ¿Cuántas clases diferentes hay en un lenguaje?
- Para el lenguaje de las palíndromes:
  - Necesitamos una *z* para clasificar a cada cadena con respecto a *pal*
  - Hay un número infinito de cadenas y para cada una de ellas podemos construir una palindrome
  - Hay *n* clases de cadenas con respecto a pal, cada una de ellas con una sóla cadena!

Los lenguajes aceptados por FA