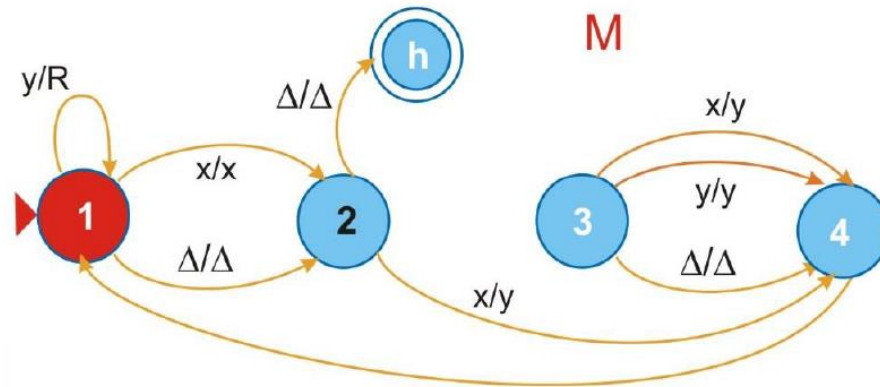
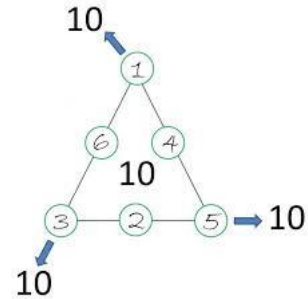


AUTOMATAS Y LENGUAJES FORMALES



INTRODUCCIÓN Y CONCEPTOS PRELIMINARES

ANTECEDENTES



Matemáticas:

Fundamentos de la lógica matemática



Ingeniería eléctrica:
Teoría de la conmutación



Lingüística:
Teórica de las gramáticas



Biología:
Redes Neuronales Artificiales

QUE ES?

La teoría de los autómatas y lenguajes formales estudia las propiedades de los modelos matemáticos de la computación.

La teoría de autómatas estudia las máquinas o dispositivos abstractos con capacidad de computación

Jerarquía de Chomsky

Lenguajes Formales	Autómatas
No enumerables	
Recursivamente enumerables	Máquina de Turing
Sensibles al contexto	Autómata Lineal Acotado
Libres de contexto	Autómata de Pila
Regulares	Autómata Finito

APLICACIONES

Compiladores

Procesadores de texto

Motores de búsqueda

Diseño de hardware

Diseño de lenguajes de programación

Teoría de conjuntos

El fundamento más importante para el estudio de los lenguajes y autómatas es la Teoría de conjuntos.

- Cantor G, Formulo la teoría de conjuntos
- Frege G, demostró que cualquier enunciado de las matemáticas puede expresarse en la teoría de conjuntos
- Russel B, Formulo la paradoja de Russel (demuestra que la teoría original de conjuntos formulada por Cantor y Frege es contradictoria.)

Conjunto

Un *conjunto* es un grupo de objetos representado como una unidad

Elemento

- Los objetos pertenecientes a un conjunto son llamados *elementos* o *miembros*.

Expresión de conjuntos

Los conjuntos pueden expresarse de dos maneras básicamente:

En extensión: Donde se citan explícitamente cada uno de sus elementos.

Ejemplo: $\{1, 3, 5\}$

En intensión: dando una descripción precisa de los elementos que forman parte del conjunto, en vez de citarlos explícitamente

Ejemplo: $\{i \in \mathbf{N} \mid \text{impar}(i), i < 6\}$

Membresía a un conjunto

Se utilizan los símbolos \in y \notin para denotar membresía o no membresía a un conjunto.

Cuando un elemento pertenece a un conjunto, se escribe el símbolo \in entre el elemento y el conjunto. Cuando un elemento no pertenece a un conjunto, se escribe el símbolo \notin entre el elemento y el conjunto.

Ejemplo:

$$6 \in \{8, 12, 6, 20\} \text{ y } 3 \notin \{8, 12, 6, 20\}$$

Subconjunto

Teniendo en cuenta los conjuntos A y B se dice que el conjunto B es subconjunto del conjunto A si B «está contenido» dentro de A, es decir si todo elemento que es miembro de B también es miembro de A. Denotado por $A \subseteq B$. Recíprocamente, se dice que el conjunto A es un superconjunto de B cuando B es un subconjunto de A.

Se dice que A es un subconjunto propio de B denotado por $A \subset B$ si A es subconjunto de B y no es igual a B

Conjunto potencia

El conjunto potencia de un conjunto dado es otro conjunto formado por todos los subconjuntos del mismo.

Ejemplo: el conjunto potencia de $A = \{1, 2, 3\}$ es:

$$\{ \emptyset, \{1\}, \{2\}, \{3\}, \{1, 2\}, \{2, 3\}, \{1, 3\}, \{1, 2, 3\} \}$$

El conjunto potencia de A también se denomina **conjunto de las partes** de A , o **conjunto de partes** de A se denota por $P(A)$ o 2^A .

Multiconjunto

Los multiconjuntos son colecciones donde se permite la aparición repetida de los elementos sin importar su orden. Se debe indicar *cuántas veces* el elemento es miembro del conjunto.

Ejemplo:

Multiconjunto $\{ c, c, c, d, d, e \}$,

Las multiplicidades de los miembros c , d , y e son 3, 2, y 1, respectivamente.

Para evitar confusión debiera escribirse: $\langle c, c, c, d, d, e \rangle$ ¹

Conjunto infinito

Un conjunto infinito es un conjunto que contiene un número infinito de elementos.

Ejemplo:

Los números enteros $Z = \{..., -3, -2, -1, 0, 1, 2, 3, ...\}$ forman un conjunto infinito y numerable.

Los puntos en una recta, representados por un [número real](#), forman un conjunto infinito y [no numerable](#)

Conjunto vacío

Un conjunto con cero elementos se denomina *conjunto vacío* y se denota por \emptyset

Singleton

- (Instancia Única). El conjunto con un solo elemento se denomina *singleton*

Cardinalidad

La **cardinalidad** de un conjunto es el número de elementos que posee ese conjunto.

El símbolo que representa la cardinalidad de un conjunto A es $n(A)$, $\text{card}(A)$, $\#A$ o $|A|$.

Ejemplo:

$A = \{-3, 0, 1, 2, 3\}$ tiene cinco elementos por tanto se tiene que $|A| = 5$

Notación Inductiva de Conjuntos

En ocasiones los conjuntos se escriben de acuerdo con una regla. Se escribe:

$\{n \mid \text{regla acerca de } n\}$

De modo que:

$\{n \mid n = m^2 \text{ para toda } m \in \mathbb{N}\}$ es el conjunto de los cuadrados perfectos

Secuencias

Una secuencia es una lista de objetos puestos en algún orden. Usualmente se denota a una secuencia por la lista de estos objetos entre paréntesis.

Ejemplo: $(-3, 0, 1, 2, 3)$

El orden no es importante en los conjuntos pero si en las secuencias de modo que $(-3, 0, 1, 2, 3) \neq (2, 0, 1, -3, 3)$

Igual pasa con las repeticiones, en los conjuntos no es tomada en cuenta pero en las secuencias si, de modo que $(-3, 0, 1, 2, 3) \neq ((-3, 0, 1, 1, 1, 2, 3)$

Tuplas

Una tupla es una lista ordenada de elementos.

Una n -tupla es una secuencia (o lista ordenada) de n elementos, siendo n un número natural (entero no-negativo). La única 0-tupla es la secuencia vacía. Una n -tupla se define inductivamente desde la construcción de un par ordenado. Las tuplas suelen anotarse listando sus elementos entre paréntesis " $()$ ", separados por comas.

Ejemplo:

$(2, 7, 4, 1, 7)$ denota una 5-tupla.

Función

Una función es un objeto que establece una relación entrada – salida. Una función toma una entrada y genera una salida. En toda función la misma entrada siempre produce la misma salida.

Si f es una función cuyo valor de salida es x^3 cuando su valor de entrada es x se escribe,

$$f(x) = x^3$$

Una función se conoce también como mapeo y si, $f(x) = x^3$ se dice que f mapea x a x^3

Argumentos de una función

Cuando el dominio de una función f es $A_1 \times A_2 \times A_3 \dots \times A_k$ para algunos conjuntos $A_1, A_2, A_3 \dots, A_k$ la entrada de f es una K -tupla $(a_1, a_2, a_3 \dots, a_k)$ y se dice que a_i son los argumentos de la función

Aridad de una función

- Una función con K elementos f es llamada función K -aria y K es llamada *aridad* de la función. Si K es 1, f tiene un solo argumento y f es llamada función unaria si K es 2, f es una función binaria

Relación

Un predicado cuyo dominio es el conjunto de las K -tuplas $A_1 \times A_2 \times A_3 \dots \times A_k$ es llamado relación k -aria. Un caso común, la relación 2-aria, es llamada relación binaria. Por lo general, se emplea la notación infija para expresar enunciados matemáticos que incluyen relaciones binarias

Relación de equivalencia

Una relación binaria es una relación de equivalencia si R satisface tres condiciones:

1. R es *reflexiva* si para toda x , xRx
2. R es *simétrica* si para toda x y y , xRy y si solo si yRx ; y
3. R es *transitiva* si para toda x , y y z , xRy y yRz implica que xRz

LENGUAJES Y EXPRESIONES REGULARES

Alfabeto

Un *alfabeto* es cualquier conjunto finito, no vacío. Los miembros de un alfabeto son llamados símbolos del alfabeto.

Es usual utilizar letras griegas mayúsculas como Σ y Γ para denotar un alfabeto.

Ejemplo:

$\{0, 1\}$,

$\{a, b, c\}$,

$\{0, 11\}$,

$\{(0, 0), (0, 1), (1, 0), (1, 1)\}$.

String

Un String sobre un alfabeto es una secuencia finita de símbolos de un alfabeto, usualmente escritos uno seguido de otros sin ser separados por comas.

Ejemplo: "abc", "invierno", "123456"

Longitud String

Si ω es un string sobre Σ , la longitud de ω , denota por $|\omega|$, es el número de símbolos que este contiene.

Ejemplo: "abc" = longitud $|3|$, "invierno" longitud $|8|$

String vacío

El string de longitud cero es llamado *String vacío* y se denota por ε

Notación para String

- Si ω tiene longitud n , puede escribirse $\omega = \omega_1, \omega_2, \omega_3, \dots, \omega_n$ donde cada $\omega_i \in \Sigma$

Reversa de un String

La reversa de ω , denotado por ω^R , es el string que se obtiene al escribir ω en el orden opuesto.

Substring

- El string z es un substring de ω si z aparece consecutivamente dentro de ω

Lenguaje

Un lenguaje es un conjunto de String

Palabra o cadena

- Es una secuencia de símbolos del alfabeto, es decir, $s = a_1a_2a_3... a_n$, donde $a_i \in \Sigma$.
- Por lo general se utilizan las primeras letras del alfabeto, a, b, c, d, e, para denotar símbolos del alfabeto y las últimas letras, s, t, u, v, w, x, y, z, para denotar palabras.

Longitud de una palabra

Es el número de símbolos en s . Se denota por $|s|$

Palabra nula o vacía Λ

- Es la palabra de longitud cero. Algunos autores utilizan ϵ para denotarla

Operaciones regulares

Las operaciones regulares se emplean en el estudio de las propiedades de los lenguajes regulares

Unión

Concatenación

Estrella

Unión

Sean A y B lenguajes. La operación regular *Unión* se define como:

$$A \cup B = \{x | x \in A \vee x \in B\}$$

Ejemplo: $A = \{1, 2, 3\}$ y $B = \{3, 4\}$

$$A \cup B = \{1, 2, 3, 4\}$$

Observación

- La unión es el lenguaje que se obtiene al combinar todos los string de A y B en un solo lenguaje

Concatenación

Sean A y B lenguajes. La operación regular concatenación se define como:

$$A \circ B = \{xy \mid x \in A \wedge y \in B\}$$

Observación

- La concatenación yuxtapone un string de B después de un string de A en todas las formas posibles para obtener los string del nuevo lenguaje

Estrella de Kleene

Sean A y B lenguajes. La operación regular *estrella* se define como:

$$A^* = \{x_1 x_2 x_3 \dots x_k \mid k \geq 0, x_i \in A\}$$

Observación

La estrella yuxtapone cualquier número de strings de A, uno después de otro, para obtener los string del nuevo lenguaje.

Ejemplo

Sea Σ el alfabeto estándar $\{a, b, c, d, e, \dots, z\}$. Si

$A = \{\text{good}, \text{bad}\}$ y $B = \{\text{boy}, \text{girl}\}$, entonces

$A \cup B = \{\text{good}, \text{bad}, \text{boy}, \text{girl}\}$

$A \circ B = \{\text{goodboy}, \text{goodgirl}, \text{badboy}, \text{badgirl}\}$

$A^* = \{\epsilon, \text{good}, \text{bad}, \text{goodgood}, \text{goodbad}, \text{badgood}, \text{badbad}, \text{goodgoodgood}, \text{goodgoodbad}, \text{goodbadgood}, \text{goodbadbad}, \dots\}$

Expresiones regulares

Las ER son simplemente fórmulas cuyo propósito es representar cada una de ellas un lenguaje. Así, el significado de una ER es simplemente el lenguaje que ella representa.

Por ejemplo, la ER “ \emptyset ” representa el conjunto vacío {}.

Equivalencias de Expresiones Regulares

Las expresiones regulares no representan en forma única a un lenguaje, esto significa que la función $L : ER \rightarrow 2^{\Sigma^*}$ no es inyectiva. Teniendo en cuenta lo anterior puede haber varias ER para un mismo lenguaje, lo cual desde luego no es conveniente, pues al ver dos ER distintas no podemos estar aún seguros que representan dos lenguajes distintos.

Ejemplo: Las siguientes ER son distintas pero representan el mismo lenguaje.

$$(a+b)^* \quad \text{Y} \quad (a^*b^*)^*$$

Sin embargo, en algunos casos resulta útil aplicar ecuaciones de equivalencia entre las ER, que son expresiones de la forma $ER_1 = ER_2$, cuyo significado es que el lenguaje de ER_1 es el mismo que el de ER_2 (contienen las mismas palabras).

Ejemplo:

La equivalencia $R + S = S + R$

Quiere decir que la suma de expresiones regulares es conmutativa, por lo que si tenemos dos ER específicas, como a^* y b^*ab , entonces la ER a^*+b^*ab será equivalente a la ER b^*ab+a^* , y ambas representan las mismas palabras.

Precedencia de los operadores:

Un operador es un símbolo que especifica el tipo de cálculo matemático que se desea realizar. Si una expresión contiene varios operadores, se realiza dichos cálculos en un orden predeterminado.

1. ()
2. * cierre
3. . concatenación
4. + unión

Propiedades de equivalencias de ER

1. Asociatividad de la unión : $\alpha + (\beta + \gamma) = (\alpha + \beta) + \gamma$

Ejemplo: $a^* + (bb + ba^*) = (a^* + bb) + ba^*$

2. Conmutatividad de la unión : $\alpha + \beta = \beta + \alpha$

Ejemplo: $a^* + bb = bb + a^*$

3. Asociatividad de la concatenación : $(\alpha\beta)\gamma = \alpha(\beta\gamma)$

Ejemplo: $(a^*bb)ba^* = a^*(bbba^*)$

4. Distributividad de la concatenación respecto de la unión :

$$\alpha(\beta + \gamma) = \alpha\beta + \alpha\gamma$$

Ejemplo: $a^*(ba + bb) = a^*ba + a^*bb$

$$(\beta + \gamma)\alpha = \beta\alpha + \gamma\alpha$$

Ejemplo: $(ba + bb)a^* = baa^* + bba^*$

5. Elemento neutro de la concatenación : $\alpha\lambda = \lambda\alpha = \alpha$

Ejemplo: $aa^*b\lambda = \lambda aa^*b = aa^*b$

6. Elemento neutro de la unión : $\alpha + \emptyset = \emptyset + \alpha = \alpha$

Ejemplo: $bab^* + \emptyset = \emptyset + bab^* = bab^*$

7. $\lambda^* = \lambda$

8. $\emptyset\alpha = \alpha\emptyset = \emptyset$

Ejemplo: $\emptyset(aba^* + bb)^* = (aba^* + bb)^*\emptyset = \emptyset$

9. $\emptyset^* = \lambda$

10. $\alpha^*\alpha^* = \alpha^*$

Ejemplo: $(a + bb)^*(a + bb)^* = (a + bb)^*$

11. $\alpha\alpha^* = \alpha^*\alpha$

Ejemplo: $(aba + b)(aba + b)^* = (aba + b)^*(aba + b)$

12. $(\alpha^*)^* = \alpha^*$

Ejemplo: $((bb + ba^*b)^*)^* = (bb + ba^*b)^*$

$$13. \alpha^* = \lambda + \alpha + \alpha^2 + \dots \alpha^n + \alpha^{n+1} \alpha^*$$

$$\text{Ejemplo: } (a + bab^*)^* = \lambda + (a + bab^*) + (a + bab^*)^2 + \dots + (a + bab^*)^n + (a + bab^*)^{n+1} (a + bab^*)^*$$

$$14. \alpha^* = \lambda + \alpha \alpha^*$$

$$\text{Ejemplo: } (a + bab^*)^* = \lambda + (a + bab^*)(a + bab^*)^*$$

$$15. \alpha^* = (\lambda + \alpha)^{n-1} + \alpha^n \alpha^*$$

$$\text{Ejemplo: } (a + bab^*)^* = (\lambda + a + bab^*)^{n-1} + (a + bab^*)^n (a + bab^*)^*$$

16. Sea $f : E_{\Sigma}^n \rightarrow E_{\Sigma}$ una composición de operaciones racionales (unión, concatenación y clausura)

$$f(\alpha, \beta, \dots, \gamma) + (\alpha + \beta + \dots + \gamma)^* = (\alpha + \beta + \dots + \gamma)^*$$

Ejemplo: Tomemos $f(a^*, bb, bab^*) = (a^*bab^* + bb)^*$. Entonces se cumple la siguiente equivalencia

$$(a^*bab^* + bb)^* + (a^* + bab^* + bb)^* = (a^* + bab^* + bb)^*$$

17. Sea $f : E_{\Sigma}^n \rightarrow E_{\Sigma}$ una composición de operaciones racionales (unión, concatenación y clausura)

$$(f(\alpha^*, \beta^*, \dots, \gamma^*))^* = (\alpha + \beta + \dots + \gamma)^*$$

Ejemplo: Tomemos $f(a^*, (bb)^*, (bab)^*) = a^*(bab)^* + (bb)^*$. Entonces se cumple la siguiente equivalencia

$$(a^*(bab)^* + (bb)^*)^* = (a + bb + bab)^*$$

$$18. (\alpha^* + \beta^*)^* = (\alpha^* \beta^*)^* = (\alpha + \beta)^*$$

$$\text{Ejemplo: } ((ab + b)^* + a^*)^* = ((ab + b)^* b^*)^* = (ab + b + a)^*$$

$$19. (\alpha \beta)^* \alpha = \alpha (\beta \alpha)^*$$

$$\text{Ejemplo: } (bb^* aba)^* bb^* = bb^* (ababb^*)^*$$

$$20. (\alpha^* \beta)^* \alpha^* = (\alpha + \beta)^*$$

$$\text{Ejemplo: } ((ab + ba)^* bb)^* (ab + ba)^* = (ab + ba + bb)^*$$

$$21. (\alpha^* \beta)^* = (\alpha + \beta)^* \beta + \lambda$$

$$\text{Ejemplo: } ((bb^* + a)^* bab^*)^* = (bb^* + a + bab^*)^* bab^* + \lambda$$

Ejemplo 1: de simplificación a partir de las propiedades de equivalencia

$$ER = 1*0(11*0)*0[(11*0)*0]^*$$

$$19. (\alpha\beta)^*\alpha = \alpha(\beta\alpha)^*$$

$$= 1*0(11*0)*[0(11*0)^*]^*0$$

$$= 1*(011^*)^*0[0(11*0)^*]^*0$$

$$= 1*(011^*)^*0[(011^*)^*0]^*0$$

Teniendo en cuenta la precedencia de operadores primero se halla lo que se encuentra dentro de: []

Por tal motivo se debe hallar la propiedad que aplique a esta parte, para el caso es la propiedad 19

Al aplicar la propiedad 19 quedo de esta manera

Se vuelve a aplicar la propiedad 19 dadas la circunstancia

Se sigue aplicando la propiedad 19

$$= 1*(011^*)*[0(011^*)]^*00$$

Se aplica la propiedad 19

$$= 1*[(011^*)^*0]^*(011^*)^*00$$

Se aplica la propiedad 20

$$= 1*(011^*+0)^*00$$

Se aplica la propiedad 4

$$= 1*(0(11^*+\Lambda))^*00$$

Se aplica la propiedad 14

$$= 1*(01^*)^*00$$

Se aplica la propiedad 19

$$= (1^*0)^*1^*00$$

Se aplica la propiedad 20

$$= (1+0)^*00$$

Se obtiene el resultado de Simplificación

Bibliografía

Apuntes de teorías de conjuntos: <http://www.mat.ucm.es/~arrondo/conjuntos.pdf>

Expresiones Regulares: Una forma diferente de expresar un lenguaje
http://ocw.unican.es/enseanzas-tecnicas/teoria-de-automatas-y-lenguajes-formales/material-de-clase-nuevo/nuevo/1-5_Expresiones_regulares.pdf

Expresiones Regulares
http://www.ia.urjc.es/grupo/docencia/automatas_itis/apuntes/capitulo7.pdf

Máquinas Secuenciales Autómatas y Lenguajes. Tema 4: Expresiones regulares.
http://www.ia.urjc.es/cms/sites/default/files/userfiles/file/GIC-MSAL/Tema4_ExpresionesRegulares.pdf

Bibliografía

Expresiones Regulares <http://users.dsic.upv.es/asignaturas/eui/alc/ER.pdf>

Expresiones Regulares <https://ccc.inaoep.mx/~emorales/Cursos/Automatas/ExpRegulares.pdf>

Expresiones Regulares. Introducción a la lógica
<http://www.cs.famaf.unc.edu.ar/wiki/lib/exe/fetch.php?media=intrologica:2014:class-3-handout-last.pdf>

Lenguajes y algebra de eventos regulares
[http://delta.cs.cinvestav.mx/~mcintosh/cellularautomata/Summer Research files/Algebra regul arPablo.pdf](http://delta.cs.cinvestav.mx/~mcintosh/cellularautomata/Summer_Research_files/Algebra_regu larPablo.pdf)