Caracterización de lenguajes regulares con expresiones regulares

Elvira Mayordomo

Universidad de Zaragoza

15 de octubre de 2012

Contenido de este tema

- Recordatorio de expresiones regulares (e.r.)
- Cómo convertir una e.r. en un autómata
- Cómo convertir un autómata en una e.r.

Expresiones regulares

Las expresiones regulares son las siguientes:

1. a (donde $a \in \Sigma$)
2. ϵ
3. $\emptyset$
4. $r_1 + r_2$ (donde r_1 y r_2 son expresiones regulares)
5. $r_1 \cdot r_2$ (donde r_1 y r_2 son expresiones regulares)
6. r^* (donde r es una expresión regular).

El lenguaje representado por una e.r. r es definido como sigue:

1. $L(a) = \{a\}$ (donde $a \in \Sigma$)
2. $L(\epsilon) = \{\epsilon\}$
3. $L(\emptyset) = \emptyset$
4. $L(r_1 + r_2) = L(r_1) \cup L(r_2)$ (donde r_1 y r_2 son expresiones regulares)
5. $L(r_1 \cdot r_2) = L(r_1) \cdot L(r_2)$ (donde r_1 y r_2 son expresiones regulares)
6. $L(r^*) = L(r)^*$ (donde r es una expresión regular).

Ejemplos

- $L(a^*) = \{a\}^* = \{a^n \mid n \in \mathbb{N}\}$
- $L(a + b) = \{a, b\}$
- $L((a + b)a^*) = \{a, b\} \cdot \{a\}^* = \{a^n \mid n \geq 1\} \cup \{ba^n \mid n \in \mathbb{N}\}$
- $L(01^*) = \{0\} \cdot \{1\}^* = \{01^n \mid n \in \mathbb{N}\}$
- $L((1 + 0) \cdot 10^*) = \{0, 1\} \cdot \{1\} \cdot \{0\}^* =$ $\{010^n \mid n \in \mathbb{N}\} \cup \{110^n \mid n \in \mathbb{N}\}$
- $L(\emptyset \cdot b) = \emptyset \cdot \{b\} = \emptyset$

Principal resultado

Teorema
Un lenguaje A es regular si y sólo si existe una e.r. r tal que $L(r) = A$.

- (Recordad Definición) Un lenguaje A es regular si existe un AFD (autómata determinista) M con $A = L(M)$.
- Vimos que un lenguaje A es regular si existe un AFnD (autómata no determinista) M con $A = L(M)$.

Cómo convertir una e.r. en un autómata

- Para ver la primera implicación (si existe una e.r. r tal que $L(r) = A$ entonces A es regular) ...
- ... vamos a encontrar un autómata M tal que $L(M) = L(r)$
- Es decir, vamos a convertir una e.r. en un autómata

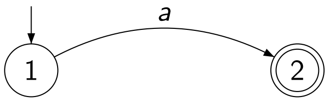
Cómo convertir una e.r. en un autómata

Vamos a seguir la definición recursiva de lenguaje representado por una e.r

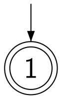
1. $L(a) = \{a\}$ (donde $a \in \Sigma$)
2. $L(\epsilon) = \{\epsilon\}$
3. $L(\emptyset) = \emptyset$
4. $L(r_1 + r_2) = L(r_1) \cup L(r_2)$ (donde r_1 y r_2 son expresiones regulares)
5. $L(r_1 \cdot r_2) = L(r_1) \cdot L(r_2)$ (donde r_1 y r_2 son expresiones regulares)
6. $L(r^*) = L(r)^*$ (donde r es una expresión regular).

- Necesitamos un autómata para $\{a\}$ (con $a \in \Sigma$)
- Necesitamos un autómata para $\{\epsilon\}$
- Necesitamos un autómata para $\emptyset$
- Luego tenemos que tratar la unión, concatenación y estrella de Kleene

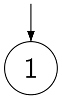
Un autómata para $\{a\}$



Un autómata para $\{\epsilon\}$



Un autómata para $\emptyset$



La unión

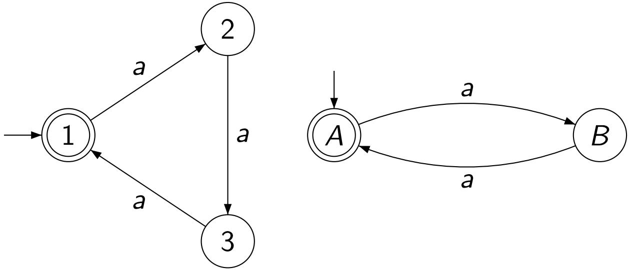
Lema

La unión de lenguajes regulares es regular.

- Tenemos que ver que si hay un autómata que reconoce A y otro que reconoce B entonces hay un autómata que reconoce $A \cup B$
- Nos basta con construir un AFnD para $A \cup B$

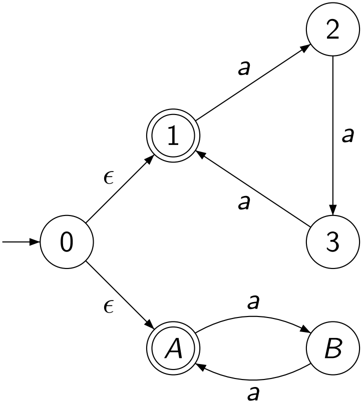
Ejemplo de unión

Por ejemplo, si tenemos los autómatas:



Ejemplo de unión

El siguiente autómata reconoce la unión de los anteriores

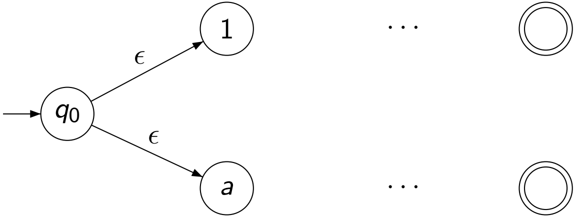


Demostración de la unión

Si tenemos un autómata que reconoce A y otro que reconoce B ...

¿Cómo hacemos un autómata que reconozca $A \cup B$?

Demostración de la unión



- Ponemos como estado inicial un estado nuevo q_0
- Añadimos ϵ-transiciones desde q_0 a los antiguos estados iniciales

La concatenación

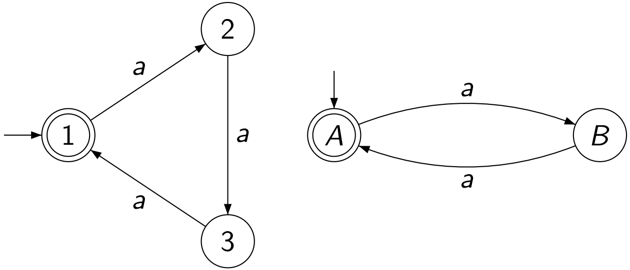
Lema

La concatenación de lenguajes regulares es regular.

- Tenemos que ver que si hay un autómata que reconoce A y otro que reconoce B entonces hay un autómata que reconoce $A \cdot B$
- Nos basta con construir un AFnD para $A \cdot B$

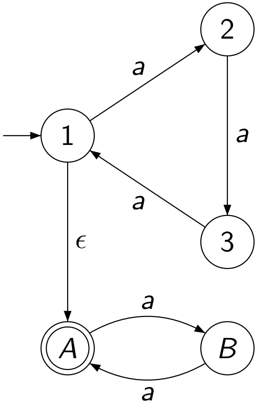
Ejemplo de concatenación

Por ejemplo, si tenemos los autómatas:



Ejemplo de concatenación

El siguiente autómata reconoce la concatenación de los anteriores

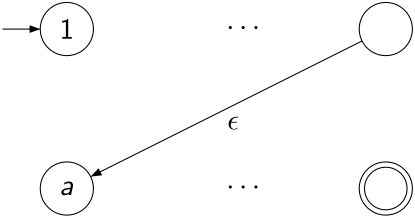


Demostración de la concatenación

Si tenemos un autómata que reconoce A y otro que reconoce B ...

¿Cómo hacemos un autómata que reconozca $A \cdot B$?

Demostración de la concatenación



- Ponemos como estado inicial el del primer autómata
- Añadimos ϵ-transiciones desde los finales del primer autómata al inicial del segundo

La estrella de Kleene

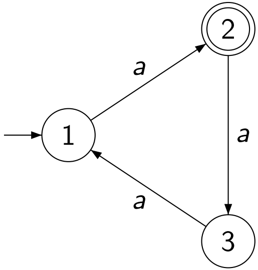
Lema

La estrella de Kleene de un lenguaje regular es regular.

- Tenemos que ver que si hay un autómata que reconoce A entonces hay un autómata que reconoce A^*
- Nos basta con construir un AFnD para A^*

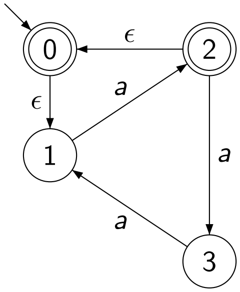
Ejemplo de estrella de Kleene

Por ejemplo, si tenemos el autómata:



Ejemplo de estrella de Kleene

El siguiente autómata reconoce la estrella de Kleene del anterior

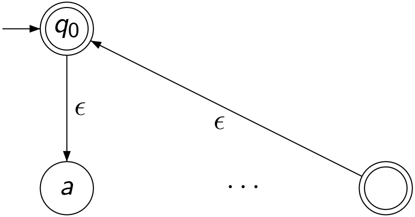


Demostración de la estrella de Kleen

Si tenemos un autómata que reconoce A ...

¿Cómo hacemos un autómata que reconozca A^*?

Demostración de la estrella de Kleene



- Ponemos como estado inicial un estado nuevo q_0
- Añadimos ϵ-transiciones desde q_0 al antiguo estado inicial
- Añadimos ϵ-transiciones desde los finales a q_0
- Hacemos q_0 final

Cómo construir un autómata equivalente a una e.r.

Ejemplo

$(00 + 111)^*$

- Primero construimos autómatas para 0 y para 1.
- Usando la construcción anterior para concatenación tenemos autómatas para 00 y para 111.
- Usando la construcción anterior para unión tenemos un autómata para $00 + 111$.
- Usando la construcción anterior para estrella de Kleene tenemos un autómata para $(00 + 111)^*$.

¿Qué hemos demostrado?

- Hemos demostrado que tenemos un autómata que reconoce los lenguajes siguientes:
  1. $L(a) = \{a\}$ (donde $a \in \Sigma$)
  2. $L(\epsilon) = \{\epsilon\}$
  3. $L(\emptyset) = \emptyset$
  4. $L(r_1 + r_2) = L(r_1) \cup L(r_2)$ (donde r_1 y r_2 son expresiones regulares)
  5. $L(r_1 \cdot r_2) = L(r_1) \cdot L(r_2)$ (donde r_1 y r_2 son expresiones regulares)
  6. $L(r^*) = L(r)^*$ (donde r es una expresión regular).
- Por tanto para cada e.r. r tenemos un autómata M tal que $L(r) = L(M)$ (es decir, M acepta el lenguaje que representa r).
- Por tanto todo lenguaje representado por una e.r. es regular.

Propiedades de clausura de los regulares

Además por el camino hemos visto:

1. La unión de dos lenguajes regulares es regular. **Los regulares son cerrados por la operación de unión.**
2. La concatenación de dos lenguajes regulares es regular. **Los regulares son cerrados por la operación de concatenación.**
3. La estrella de Kleene de un lenguaje regular es regular. **Los regulares son cerrados por la operación de estrella de Kleene.**

Otras propiedades de clausura de los regulares

1. El complementario de un lenguaje regular es regular. **Los regulares son cerrados por la operación de complemento.**
2. El reverso de un lenguaje regular es regular. **Los regulares son cerrados por la operación de reverso.**
3. La intersección de dos lenguajes regulares es regular. **Los regulares son cerrados por la operación de intersección.**

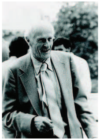
El complementario y el reverso son muy fáciles, la intersección tiene algo más de trabajo

$$\{w \in \{a, b\} \mid |w|_a = 2n, |w|_b = 2m, \quad n, m \in \mathbb{N}\} =$$

$$= \{w \in \{a, b\} \mid |w|_a = 2n, n \in \mathbb{N}\} \cap \{w \in \{a, b\} \mid |w|_b = 2m, n \in \mathbb{N}\}$$

Curiosidades: Kleene y la estrella de Kleene

- Stephen C. Kleene fue un matemático americano (1909-1994).
- En los 50 inventó las expresiones regulares.
- Muy interesado como Turing en formalizar qué es computación y encontrar los límites de un ordenador.
- Sus aportaciones tuvieron gran influencia en la programación lógica.

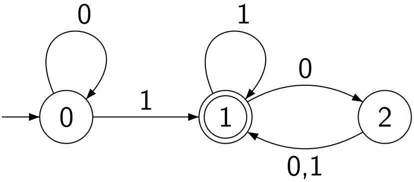


¿Qué nos falta?

- Queremos demostrar que un lenguaje A es regular si y sólo si existe una e.r. r tal que $L(r) = A$.
- Hemos visto que si existe una e.r. r tal que $L(r) = A$ entonces A es regular (es decir, hemos convertido cada e.r. en un autómata).
- Nos falta ver que si A es regular entonces existe una e.r. r tal que $L(r) = A$ (es decir, nos falta convertir cada autómata en una e.r.).
- Por la equivalencia AFDs y AFnDs nos basta hacerlo con AFDs (autómatas deterministas).

Convertir un AFD en una e.r. equivalente

- Tenemos un AFD $M = (Q, \Sigma, \delta, q_0, F)$ como este:



- Para cada estado $q \in Q$, L_q es el conjunto de palabras que aceptamos empezando en el estado q, es decir,

$$L_q = \{w \mid \delta(q, w) \in F\}$$

- Por tanto $L_{q_0} = L(M)$, el lenguaje aceptado por M.

Ecuaciones sobre un autómata

- Podemos escribir ecuaciones sencillas que relacionen los L_q separando el primer símbolo.
- Por ejemplo, L_2 son las palabras que empiezan en 0 seguidas de una palabra de L_1 y las palabras que empiezan en 1 seguidas de una palabra de L_1

$$L_2 = 0 \cdot L_1 + 1 \cdot L_1$$

(Usamos notación de e.r., "$+$" es la unión)

$$L_0 = 0 \cdot L_0 + 1 \cdot L_1$$

$$L_1 = \epsilon + 0 \cdot L_2 + 1 \cdot L_1$$

- Si conseguimos despejar L_0 tenemos quién es $L(M)$

Resolviendo las ecuaciones ...

- Sustituimos L_2 en L_1:

$$L_1 = \epsilon + 0 \cdot (0 \cdot L_1 + 1 \cdot L_1) + 1 \cdot L_1$$

- Agrupando:

$$L_1 = \epsilon + (00 + 01 + 1) \cdot L_1$$

- ¿Cómo despejamos L_1?
- Lo sabemos hacer en matemáticas: $x = 2 + 5x$
- Tenemos un lema que nos permite resolver ecuaciones de la forma $L = A \cdot L + B$, es el lema de Arden.

El lema de Arden

Lema

Si A y B son dos lenguajes con $\epsilon \notin A$ y X es un lenguaje desconocido, entonces la única solución a la ecuación $X = AX + B$ es $X = A^* B$.

La demostración en la pizarra y en la web

Resolviendo las ecuaciones ...

$$
\begin{aligned}
L_0 &= 0 \cdot L_0 + 1 \cdot L_1 \\
L_1 &= \epsilon + 0 \cdot L_2 + 1 \cdot L_1 \\
L_2 &= 0 \cdot L_1 + 1 \cdot L_1
\end{aligned}
$$

- Habíamos despejado

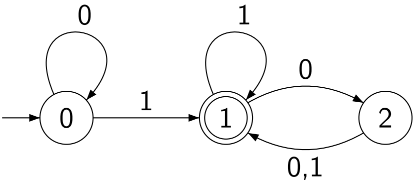
$$L_1 = \epsilon + (00 + 01 + 1) \cdot L_1$$

- Luego por el lema de Arden $L_1 = (00 + 01 + 1)^*$
- Sustituyendo en L_0:

$$L_0 = 0 \cdot L_0 + 1(00 + 01 + 1)^*$$

- Por el lema de Arden $L_0 = 0^*1(00 + 01 + 1)^*$

Fin del ejemplo



- Este autómata reconoce el lenguaje representado por $0^*1(00+01+1)^*$.

Visto en este tema

- Cómo convertir una e.r. en un autómata.
- Cómo convertir un autómata en una e.r.
- Los regulares son cerrados por unión, concatenación y estrella de Kleene (y complemento, reverso e intersección).
- Para ver que un lenguaje es regular basta con encontrar una e.r. que lo represente (o un autómata que lo acepte, determinista o no determinista).

Para comparar dos e.r.

- Recordad que sabemos comparar autómatas (hay que determinizar y minimizar)
- Para comparar dos e.r. hay que convertirlas en dos autómatas y compararlos.

Bibliografía

- Kelley, sección 2.8.
- Sipser (2a edición), páginas 44 a 47 y 58 a 69 (en secciones 1.1, 1.2 y 1.3). No incluye la demostración presentada de Cómo convertir un autómata en una e.r.