PONTIFICIA UNIVERSIDAD CATÓLICA DE CHILE

FACULTAD DE LETRAS

Introducción a la lingüística computacional

César Antonio Aguilar

Facultad de Lenguas y Letras

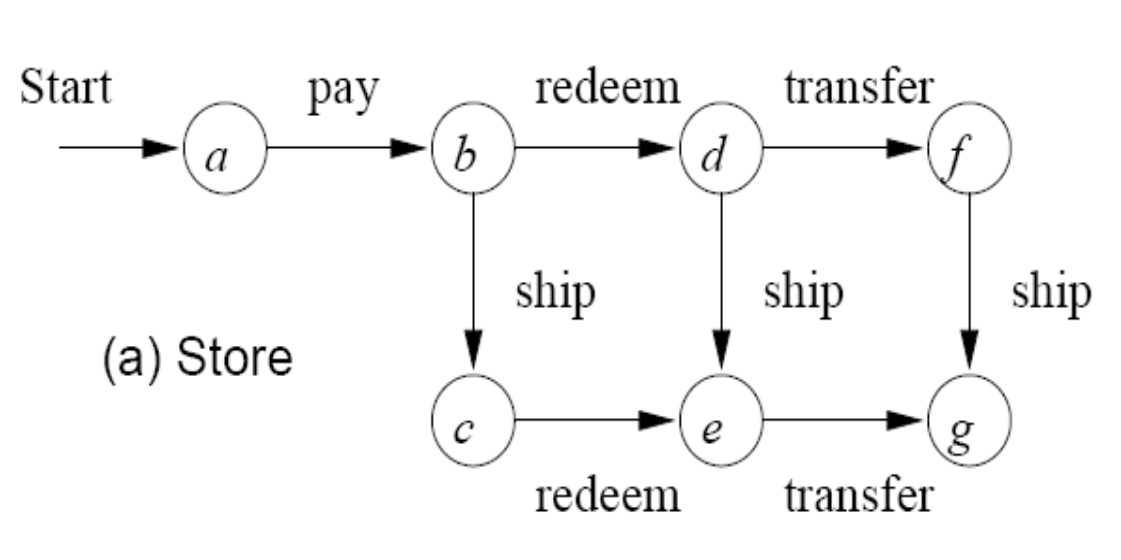
22/08/2017

Cesar.Aguilar72@gmail.com

Teoría de autómatas (1)

La **teoría de autómatas** es una línea de investigación de las ciencias computacionales, enfocada en la representación y análisis de problemas que puedan ser resueltos "computacionalmente". La teoría de autómatas tiene bases matemáticas, en particular de dos áreas:

- Teoría de conjuntos
- Modelos estadísticos (máquinas de Markov)



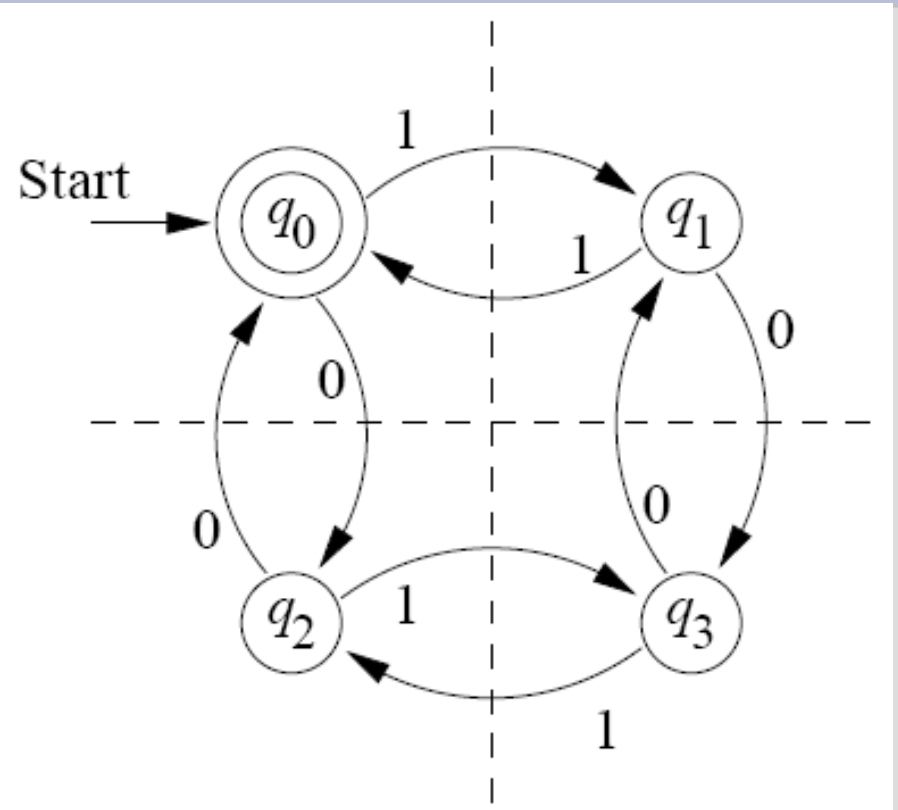
Ejemplo de un autómata diseñado para realizar transacciones bancarias

Teoría de autómatas (2)

Un autómata, a grandes rasgos, es una máquina compuesta por conjunto de estados –esto es, por una cadena de procesos–, organizados de tal manera que ofrezcan un resultado específico.

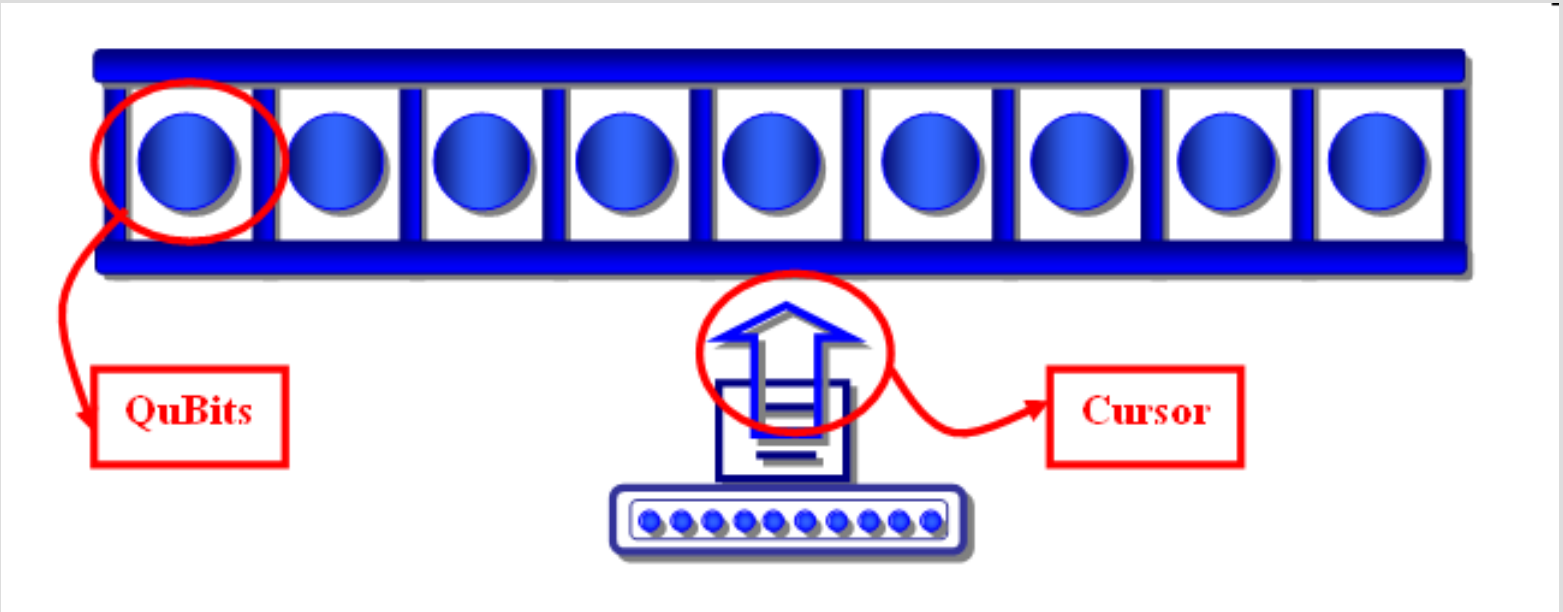
El autómata trabaja del siguiente modo: imaginemos que tenemos cierto tipo de dato que deseamos validar, p. e., una secuencia de números que conforman el código de un cliente bancario.

Supongamos que establecemos que esa secuencia se forma a partir de 8 dígitos: año de nacimiento (4 números), mes (2 dígitos) y día (2 dígitos).



Teoría de autómatas (3)

La metáfora que usualmente se ocupa para describir un autómata es verlo como un “cursor” que recorre una “cinta” con nuestros datos:



Así, el cursor solamente toma en cuenta aquellos números que son necesarios para formar la secuencia numérica “Año/Mes/Día”, y “salta” aquellos que no necesita.

Teoría de autómatas (4)

¿Cómo se construye un autómata? Necesitamos 5 elementos:

Un conjunto de estados, Q, que representan cada uno de los procesos que tenemos que cubrir para llegar a un resultado.

Un conjunto de símbolos, Σ, que agrupa todos los caracteres que conforman nuestro lenguaje L.

El signo δ representa un estado de transición, el cual mapea un estado a partir de un símbolo: $\delta = Q \times \Sigma \rightarrow Q$.

$q0$ representa el estado inicial, que es la entrada para nuestros datos, de modo que $q0 \in Q$.

F representa todos aquellos estados que cuentan con aquellas secuencias que son pertinentes, de modo que: $F \subseteq Q$.

Teoría de autómatas (5)

La forma en como trabaja el autómata es la siguiente:

Nuestro estado de inicio es aquél en donde se presenta un número, p. e. *1*. Inmediatamente, podemos establecer 3 estados para construir las siguientes secuencias:

Año: necesitamos 4 números, y que nuestra secuencia empiece con 1.

Mes: necesitamos 2 número, en donde sólo vamos admitir aquellas combinaciones que vayan de 0 a 12.

Día: necesitamos 2 números, y sólo admitimos aquellas secuencias que van de 1 a 31.

Teoría de autómatas (6)

Un autómata razona, más o menos, del siguiente modo:

Supongamos que ingresamos el número *1* en nuestra entrada.

El autómata entiende que $1 \in \Sigma$.

La función recibe *1* en un estado *Q*, y pasa a otro estado Q, es decir $\delta_1: Q \rightarrow Q$

1 va a ser un argumento que cubre cierta función q: $\delta_1(q)$, y $q \in Q$

Podemos entonces decir que *1* nos sirve para escribir una cadena *año*

El habla regular de los corderos (1)

¿Cómo platican las ovejas entre ellas? Veamos:

Podemos decir que sus "oraciones" se componen de:

1. Consonante oclusiva sorda
2. Secuencia de vocales abiertas
3. Énfasis tonal para señalar el término de una emisión

El habla regular de los corderos (2)

Desarrollemos un autómata para este lenguaje:

Existe un alfabeto Σ, que tiene tres símbolos: {*b, a,* !}.

q0 es el estado inicial.

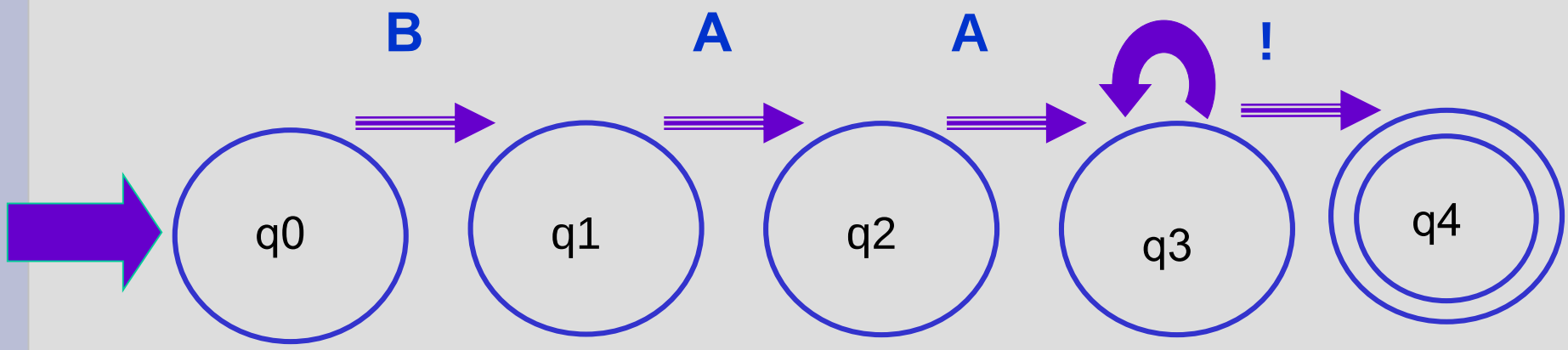
δ indica las transiciones:
tenemos 5.

q4 será un estado de aceptación, esto es, el que señala el punto final de nuestro proceso.

Un conjunto de estados, *Q*, que representan cada uno de los procesos que tenemos que cubrir para llegar a un resultado.

El habla regular de los corderos (3)

Hay una forma de representar esto gráficamente:



Según el autómata, podemos generar cadenas del tipo:

Baaa!
Baaaa!
Baaaaa!
Baaaaaa!

Baaaaaaa!
Baaaaaaaa!
Baaaaaaaaa!
Baaaaaaaaaa!

El habla regular de los corderos (4)

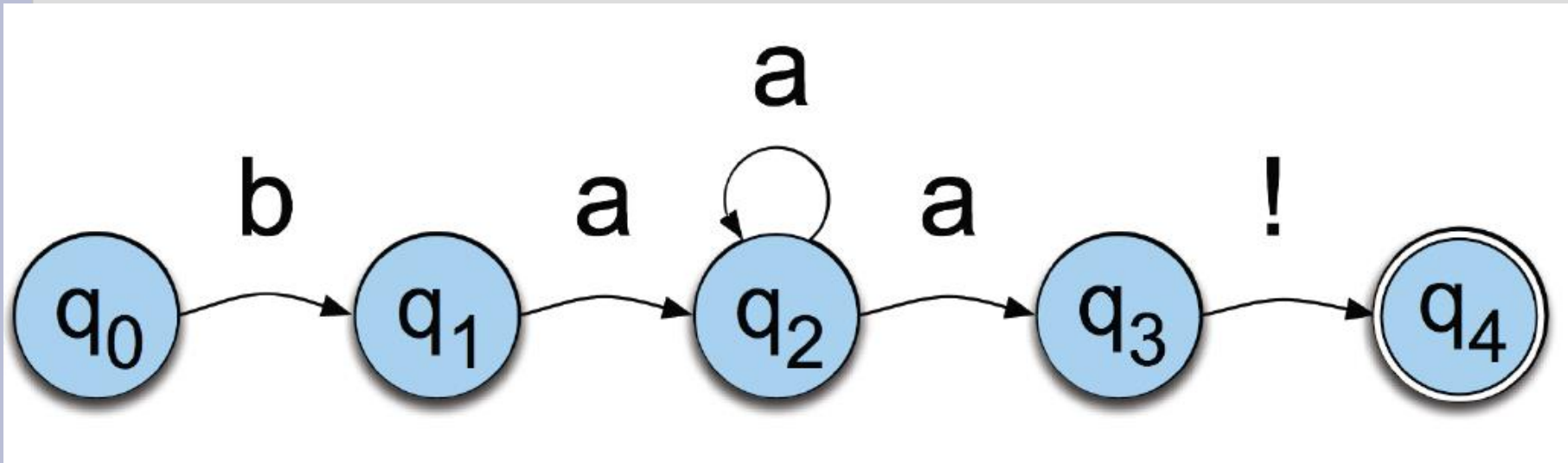
Símbolos/Estados	B	a	a	a	!
q0	1				
q1		2			
q2			3	3	
q3					4

Para darnos una idea sobre cómo va a trabajar nuestro autómata, podemos implementar una *tabla de transición*.

Una tabla de transición permite relacionar la transición de estado que le corresponde a cada símbolo de nuestro alfabeto.

El habla regular de los corderos (5)

Otro autómata posible es el siguiente:

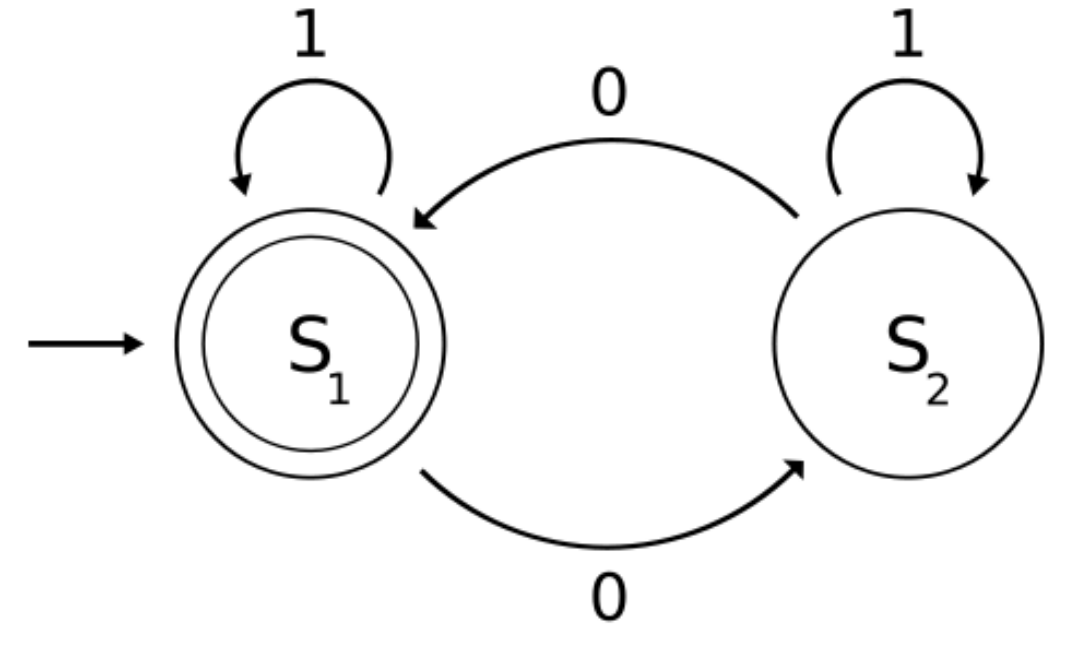


¿Cuál es la diferencia? Que el primero es un autómata finito determinista, y éste es un autómata finito no determinista.

Dos clases de autómatas (1)

Teoría de autómatas finitos deterministas:

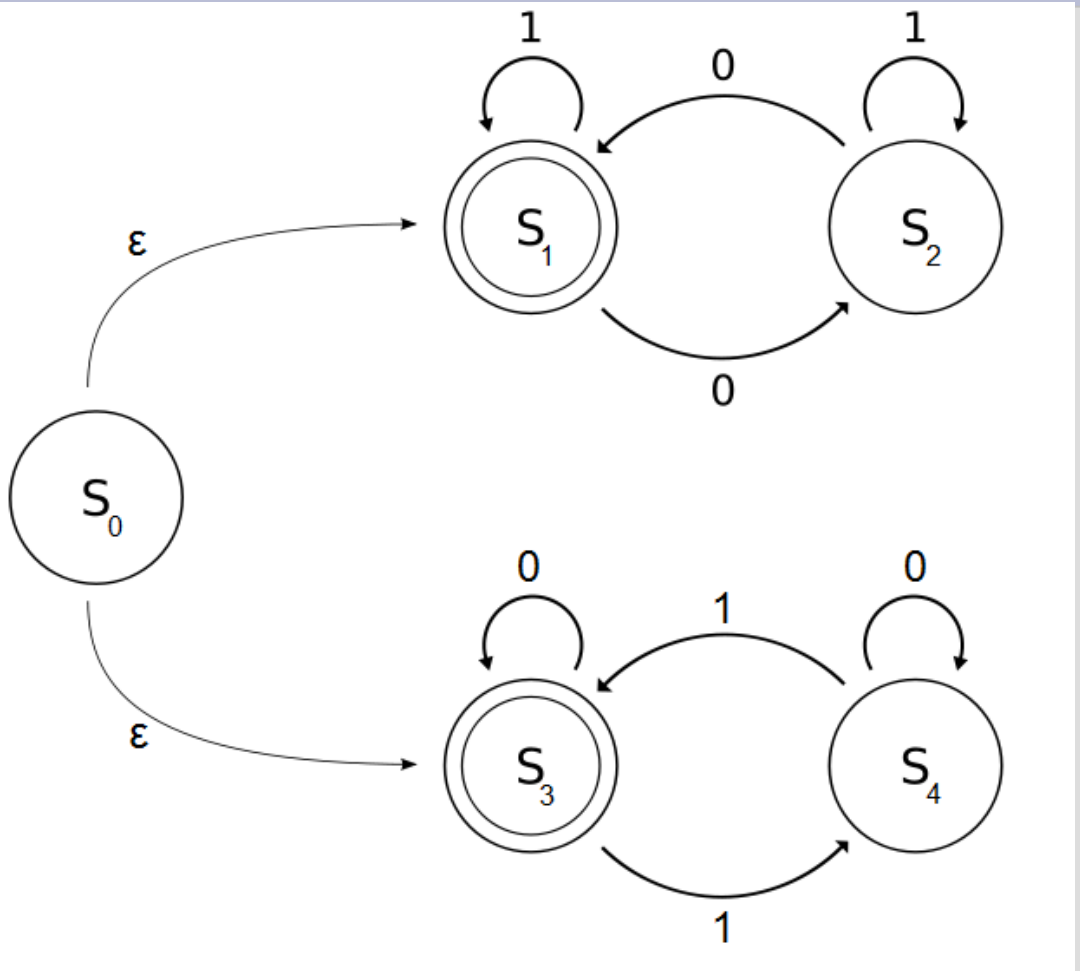
Un AFD es un autómata que establece una sola transición para cada símbolo del alfabeto.



Dos clases de autómatas (2)

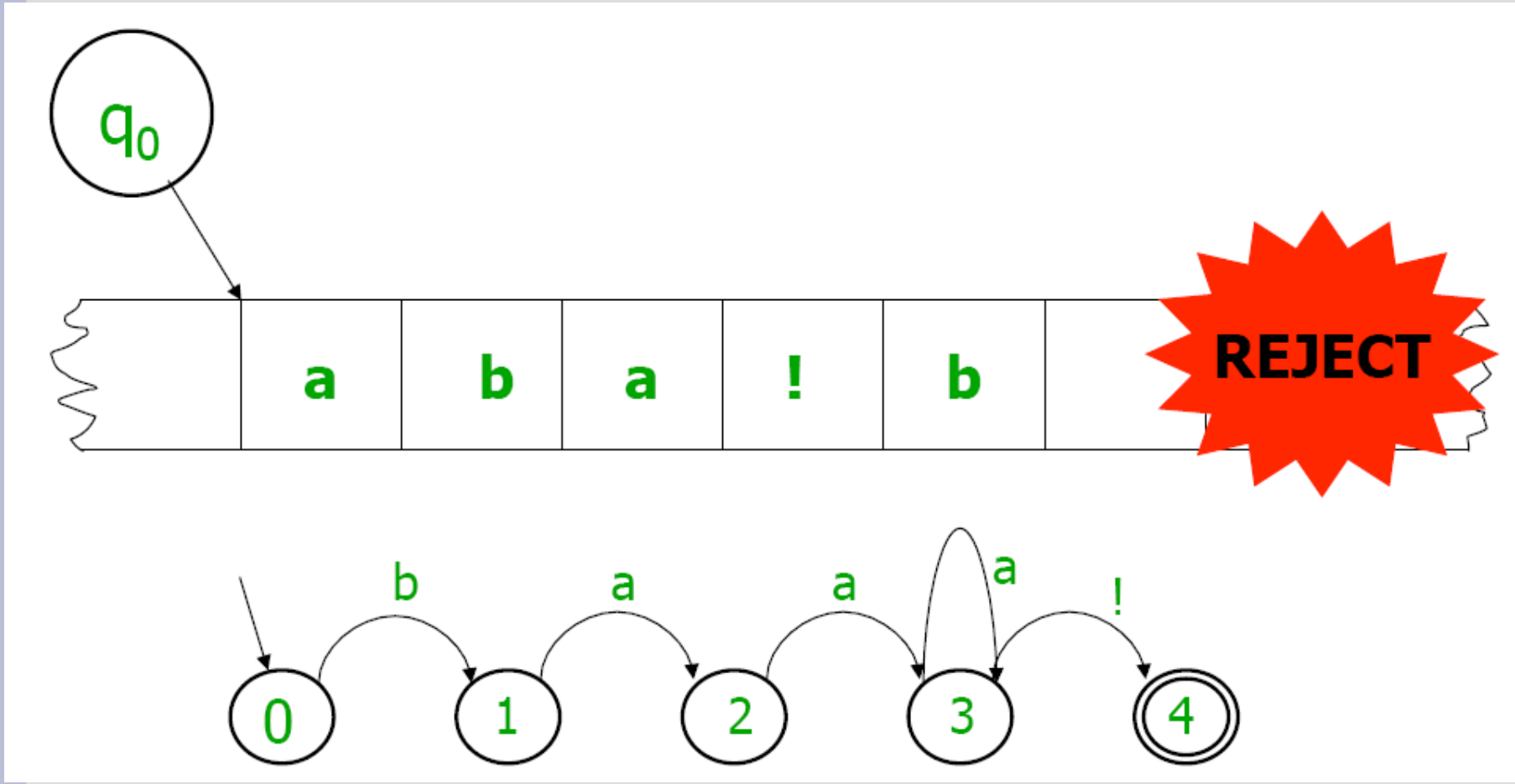
Teoría de autómatas finitos no deterministas:

Un AFND es un autómata que establece una o varias transiciones para uno o varios símbolos del alfabeto.



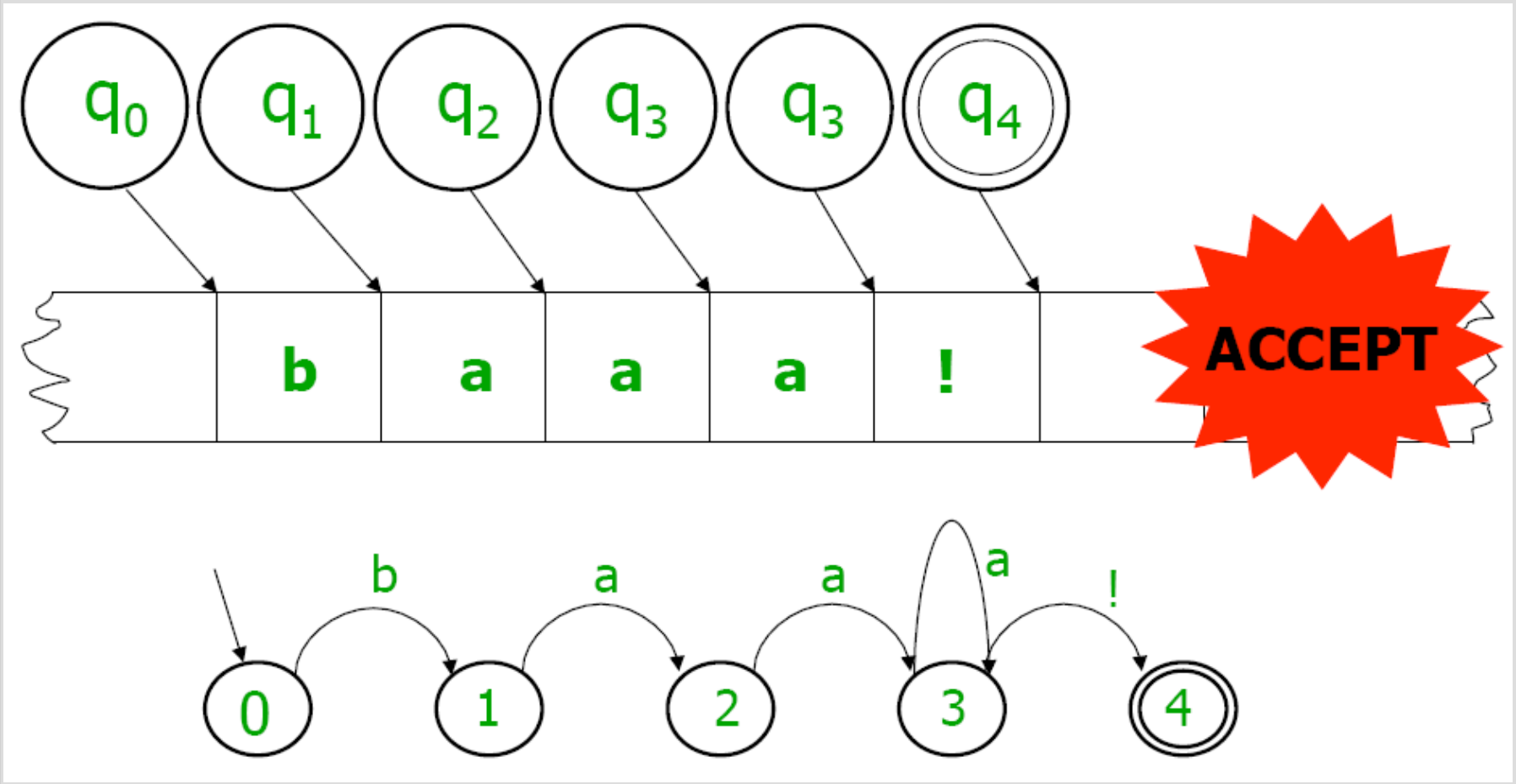
Resultados (1)

Nuestro autómata permite filtrar cadenas mal construidas como:



Resultados (2)

Igualmente, es capaz de validar construcciones como:



Tarea (1)

1. Elaborar un autómata, el cual puede ser determinista o no-determinista.
2. Vamos a validar correos que terminen en @iingen.unam.mx. Estos correos muestran el siguiente patrón:

   - Primera letra mayúscula del primer nombre.
   - Primer apellido completo en letras minúsculas.
   - Primera letra mayúscula del segundo apellido.
   - Año de nacimiento, p. e.: 1972, 1982, 1992, etc.
   - Terminación @iingen.unam.mx.

Tarea (2)

1. AhernandezA1981@iingen.unam.mx
2. OacostaL1969@iingen.unam.mx
3. OsanchezV1978@iingen.unam.mx
4. AreyesP1976@iingen.unam.mx
5. Jsmith1950@iingen.unam.mx
6. Jleclezio1946@iingen.unam.mx
7. Aruyperez1973@iingen.unam.mx
8. IruyP1976@iingen.unam.mx
9. LmejíaL1983@iingen.unam.mx
10. Fgoncalez1988@iingen.unam.mx
11. Jvang1941@iingen.unam.mx
12. Acardero1958@iingen.unam.mx
13. AladrondeguevaraR1972@iingen.unam.mx
14. GWilliamIII1945@iingen.unam.mx
15. Munamuno1920@iingen.unam.mx

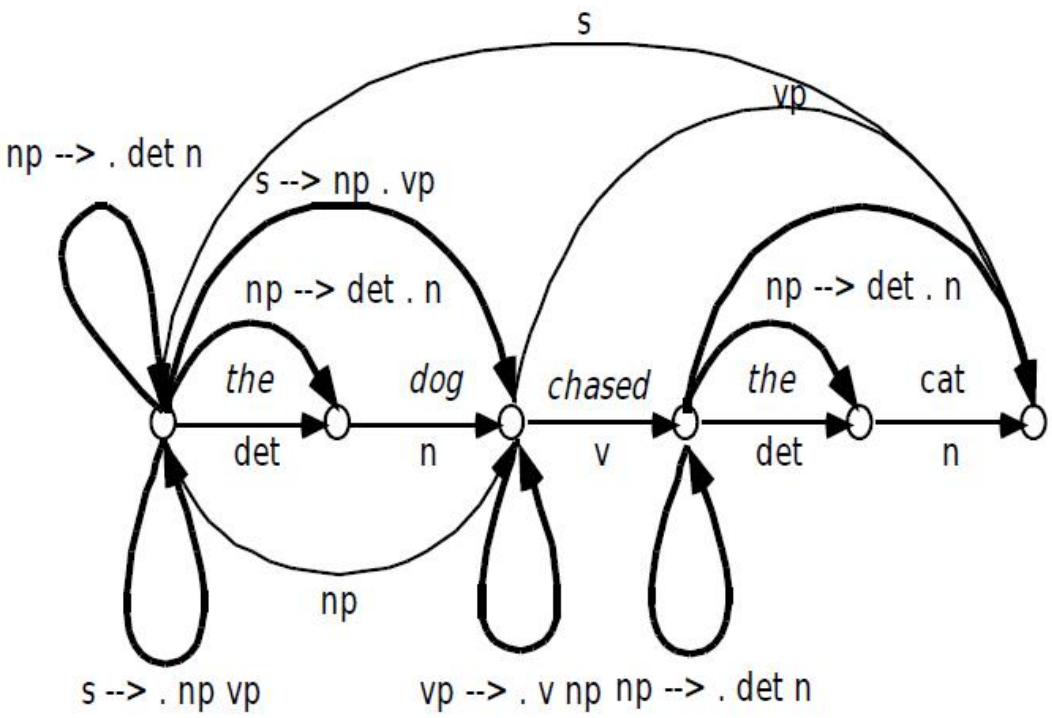
¿Y todo esto…, para qué? (1)

- Vistos así, los lenguajes formales son la cosa más simple y aburrida del mundo.
- Al final, Chomsky y otros lingüistas han demostrado (con enormes cantidades de datos) que las lenguas naturales NO SE COMPORTAN EXACTAMENTE IGUAL que los lenguajes artificiales.
- Entonces: ¿para qué diablos me sirve saber qué es un lenguaje formal?
- Siendo menos “mala leche”, ¿puedo sacarle provecho a esto?

La respuesta es: “Sí”.

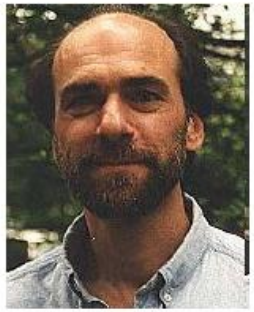
La venganza de los Nerds… (por lo menos en LC)

En cómputo, las ideas de Chomsky permitieron el desarrollo de sistemas para el tratamiento del lenguaje natural.



Eugene Charniak

Aravind K. Joshi



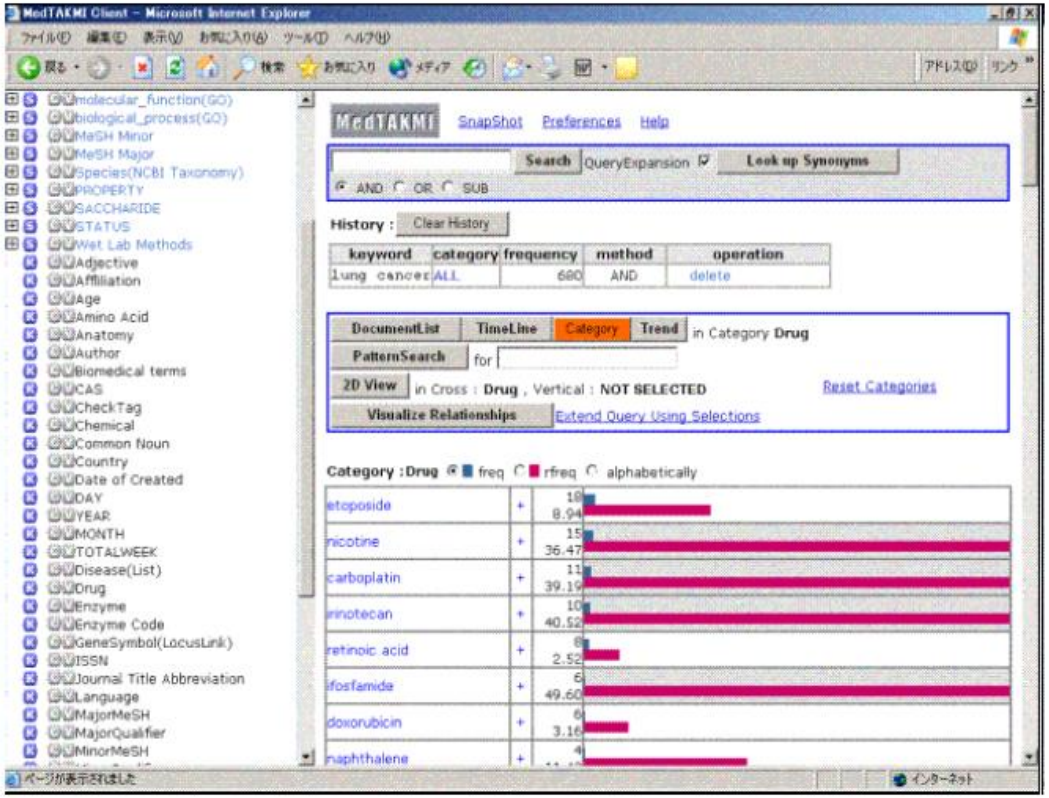
James Pustejovsky

Ivan Sag

Hacia las expresiones regulares (1)

¿Qué son las expresiones regulares?

Una expresión regular es un tipo de patrón o estructura que describe un conjunto de cadenas sin enumerar sus elementos. Es “regular porque su codificación tiende a ser repetitiva (de hecho, se pueden considerar una especie de “rutina” para comunicar cierta información).



Hacia las expresiones regulares (2)

En PLN, nos ayudan a procesar grandes repositorios de documentos, conocidos como *corpus lingüísticos.*

Gracias por su atención

Blog del curso:
http://cesaraguilar.weebly.com/introduccioacuten-a-la-linguumliacutestica-computacional.html