Conceptos preliminares

José de Jesús Lavalle Martínez

Benemérita Universidad Autónoma de Puebla
Facultad de Ciencias de la Computación
Lenguajes Formales y Autómatas CCOS 014

Primavera 2021

Contenido

Motivación general

- Tres ideas fundamentales son los temas principales de este curso: lenguajes, gramáticas y autómatas.

Motivación general

- En el transcurso de nuestro estudio, exploraremos muchos resultados sobre estos conceptos y sobre su relación entre sí.

Motivación general

- Pero primero, debemos comprender el significado de los términos.

Motivación Lenguajes

- Todos estamos familiarizados con la noción de lenguajes naturales, como el español, el inglés y el francés.

Motivación Lenguajes

- Aún así, para la mayoría de nosotros probablemente resultará difícil decir exactamente lo que significa la palabra “lenguaje”.

Motivación Lenguajes

- Los diccionarios definen informalmente el término como un sistema adecuado para la expresión de ciertas ideas, hechos o conceptos, incluyendo un conjunto de símbolos y reglas para su manipulación.

Motivación Lenguajes

- Si bien esto nos da una idea intuitiva de lo que es un lenguaje, no es suficiente como definición para el estudio de los lenguajes formales.

Motivación Lenguajes

- Necesitamos una definición precisa del término.

- Comenzamos con un conjunto Σ, finito y no vacío, de símbolos, llamado **alfabeto**.

Alfabeto y cadena

- A partir de los símbolos individuales, construimos **cadenas**, que son secuencias finitas de símbolos del alfabeto.

Alfabeto y cadena

- Por ejemplo, si el alfabeto $\Sigma = \{a, b\}$, entonces $abab$ y $aaabbba$ son cadenas en Σ.

- Con pocas excepciones, usaremos letras minúsculas $a, b, c, \ldots$ para elementos de Σ y $u, v, w, \ldots$ para nombres de cadenas.

- Escribiremos, por ejemplo,

$$w = abaaa$$

para indicar que la cadena denominada w tiene el valor específico $abaaa$.

Concatenación y reversa

- La **concatenación** de dos cadenas w y v es la cadena que se obtiene añadiendo los símbolos de v al extremo derecho de w,

- es decir, si

$$w = a_1 a_2 \cdots a_n$$

y

$$v = b_1 b_2 \cdots b_m,$$

entonces la concatenación de w y v, denotada por wv, es

$$wv = a_1 a_2 \cdots a_n b_1 b_2 \cdots b_m.$$

Concatenación y reversa

- La **reversa** de una cadena se obtiene escribiendo los símbolos en orden inverso; si w es una cadena como se muestra arriba, entonces su reversa w^R es
$$w^R = a_n \cdots a_2 a_1.$$

Longitud

- La **longitud** de una cadena w, denotada por $|w|$, es el número de símbolos en la cadena.

Longitud

- Con frecuencia necesitaremos hacer referencia a la cadena vacía, que es una cadena sin ningún símbolo. Será denotada por λ.

Longitud

- Las siguientes relaciones simples

$$|\lambda| = 0,$$
$$\lambda w = w\lambda = w$$

son válidas para toda w.

Subcadena, prefijo y sufijo

- Cualquier cadena de símbolos consecutivos en w es una **subcadena** de w.

Subcadena, prefijo y sufijo

- Si

$$w = vu,$$

entonces se dice que las subcadenas v y u son un **prefijo** y un **sufijo** de w, respectivamente.

- Por ejemplo, si $w = abbab$, entonces $\{\lambda, a, ab, abb, abba, abbab\}$ es el conjunto de todos los prefijos de w, mientras que bab, ab, b son algunos de sus sufijos.

Propiedades de las cadenas I

- Las propiedades simples de las cadenas, como su longitud, son muy intuitivas y probablemente necesiten poca elaboración.

Propiedades de las cadenas I

- Por ejemplo, si u y v son cadenas, entonces la longitud de su concatenación es la suma de las longitudes individuales, es decir,

Propiedades de las cadenas I

- $$|uv| = |u| + |v|. \tag{1}$$

Propiedades de las cadenas I

- Pero aunque esta relación es obvia, es útil poder precisarla y probarla. Las técnicas para hacerlo son importantes en situaciones más complicadas.

Propiedades de las cadenas II

Ejemplo 1

Demuestre que $|uv| = |u| + |v|$ se cumple para cualquier u y v.

Propiedades de las cadenas II

Ejemplo 1

Demuestre que $|uv| = |u| + |v|$ se cumple para cualquier u y v.

Para probar esto, primero necesitamos una definición de la longitud de una cadena.

Propiedades de las cadenas II

Ejemplo 1

Demuestre que $|uv| = |u| + |v|$ se cumple para cualquier u y v.

Hacemos tal definición de manera recursiva mediante

$$|a| = 1,$$
$$|wa| = |w| + 1,$$

para todo $a \in \Sigma$ y w cualquier cadena en Σ.

Propiedades de las cadenas II

Ejemplo 1

Demuestre que $|uv| = |u| + |v|$ se cumple para cualquier u y v.

Hacemos tal definición de manera recursiva mediante

$$|a| = 1,$$
$$|wa| = |w| + 1,$$

para todo $a \in \Sigma$ y w cualquier cadena en Σ.

Esta definición es una declaración formal de nuestra comprensión intuitiva de la longitud de una cadena: la longitud de un solo símbolo es uno, y la longitud de cualquier cadena aumenta en uno si le agregamos otro símbolo.

Propiedades de las cadenas III

Con esta definición formal, estamos listos para probar que $|uv| = |u| + |v|$ mediante inducción.

Propiedades de las cadenas III

Por definición, $|uv| = |u| + |v|$ se cumple para todo u de cualquier longitud y todo v de longitud 1, por lo que tenemos que la base de la inducción se cumple.

Como suposición inductiva, asumimos que $|uv| = |u| + |v|$ se cumple para todo u de cualquier longitud y todo v de longitud $1, 2, \ldots, n$.

Propiedades de las cadenas IV

Ahora tome cualquier v de longitud $n+1$ y escríbala como $v = wa$.

Propiedades de las cadenas IV

Ahora tome cualquier v de longitud $n+1$ y escríbala como $v = wa$.

Luego,

$$\begin{aligned} |v| &= |w| + 1, \\ |uv| &= |uwa| = |uw| + 1. \end{aligned}$$

Propiedades de las cadenas IV

Ahora tome cualquier v de longitud $n + 1$ y escríbala como $v = wa$.

Luego,

$$|v| = |w| + 1,$$
$$|uv| = |uwa| = |uw| + 1.$$

Por la hipótesis inductiva (que es aplicable ya que w es de longitud n),

$$|uw| = |u| + |w|$$

así que

$$|uv| = |u| + |w| + 1 = |u| + |v|.$$

Propiedades de las cadenas IV

Ahora tome cualquier v de longitud $n+1$ y escríbala como $v = wa$.

Luego,

$$|v| = |w| + 1,$$
$$|uv| = |uwa| = |uw| + 1.$$

Por la hipótesis inductiva (que es aplicable ya que w es de longitud n),

$$|uw| = |u| + |w|$$

así que

$$|uv| = |u| + |w| + 1 = |u| + |v|.$$

Por lo tanto, $|uv| = |u| + |v|$ se cumple para todo u y todo v de longitud hasta $n+1$, completando el paso inductivo y la demostración. $\square$

- Si w es una cadena, entonces w^n representa la cadena obtenida al concatenar w n veces.

- Como caso especial, definimos

$$w^0 = \lambda,$$

para toda w.

w^n, Σ^* y Σ^+ I

- Si Σ es un alfabeto, usamos Σ^* para denotar el conjunto de cadenas obtenido al concatenar cero o más símbolos de Σ.

- El conjunto Σ^* siempre contiene λ.

- Para excluir la cadena vacía, definimos

$$\Sigma^+ = \Sigma^* - \{\lambda\}.$$

w^n, Σ^* y Σ^+ II

- Mientras que Σ es finito por suposición, Σ^* y Σ^+ son siempre infinitos ya que no hay límite en la longitud de las cadenas en estos conjuntos.

- Un lenguaje se define muy generalmente como un subconjunto de Σ^*. Una cadena en un lenguaje L se llamará **oración** de L.

w^n, Σ^* y Σ^+ II

- Esta definición es bastante amplia; cualquier conjunto de cadenas de un alfabeto Σ puede considerarse un lenguaje.

- Más adelante estudiaremos métodos mediante los cuales se pueden definir y describir lenguajes específicos; esto nos permitirá estructurar este concepto bastante amplio.

w^n, Σ^* y Σ^+ II

- Por el momento, sin embargo, solo veremos algunos ejemplos específicos.

Ejemplo 2

Ejemplo 2

Sea $\Sigma = \{a, b\}$. Entonces

$$\Sigma^* = \{\lambda, a, b, aa, ab, ba, bb, aaa, aab, \ldots\}.$$

Ejemplo 2

Ejemplo 2

Sea $\Sigma = \{a, b\}$. Entonces

$$\Sigma^* = \{\lambda, a, b, aa, ab, ba, bb, aaa, aab, \ldots\}.$$

El lenguaje

$$\{a, aa, aab\}$$

es un lenguaje sobre Σ. Debido a que tiene un número finito de oraciones, lo llamamos lenguaje finito.

Ejemplo 2

Ejemplo 2

Sea $\Sigma = \{a, b\}$. Entonces

$$\Sigma^* = \{\lambda, a, b, aa, ab, ba, bb, aaa, aab, \ldots\}.$$

El conjunto

$$L = \{a^n b^n : n \geq 0\}$$

también es un lenguaje sobre Σ.

Ejemplo 2

Ejemplo 2

Sea $\Sigma = \{a, b\}$. Entonces

$$\Sigma^* = \{\lambda, a, b, aa, ab, ba, bb, aaa, aab, \ldots\}.$$

El conjunto

$$L = \{a^n b^n : n \geq 0\}$$

también es un lenguaje sobre Σ. Las cadenas $aabb$ y $aaaabbbb$ están en el lenguaje L,

Ejemplo 2

Ejemplo 2

Sea $\Sigma = \{a, b\}$. Entonces

$$\Sigma^* = \{\lambda, a, b, aa, ab, ba, bb, aaa, aab, \ldots\}.$$

El conjunto

$$L = \{a^n b^n : n \geq 0\}$$

también es un lenguaje sobre Σ. Las cadenas $aabb$ y $aaaabbbb$ están en el lenguaje L, pero la cadena abb no está en L. Este lenguaje es infinito.

Ejemplo 2

Ejemplo 2

Sea $\Sigma = \{a, b\}$. Entonces

$$\Sigma^* = \{\lambda, a, b, aa, ab, ba, bb, aaa, aab, \ldots\}.$$

El conjunto

$$L = \{a^n b^n : n \geq 0\}$$

también es un lenguaje sobre Σ. Las cadenas $aabb$ y $aaaabbbb$ están en el lenguaje L, pero la cadena abb no está en L. Este lenguaje es infinito. Los lenguajes más interesantes son infinitos. $\square$

Operaciones sobre lenguajes I

- Dado que los lenguajes son conjuntos, la unión, intersección y diferencia de dos lenguajes se definen inmediatamente.

Operaciones sobre lenguajes I

- El complemento de un lenguaje se define con respecto a Σ^*; es decir, el complemento de L es

$$\overline{L} = \Sigma^* - L$$

- La reversa de un lenguaje L es el conjunto de todas las reversas de las cadenas de L, es decir,

$$L^R = \{w^R : w \in L\}.$$

Operaciones sobre lenguajes I

- La concatenación de dos lenguajes L_1 y L_2 es el conjunto de todas las cadenas obtenidas al concatenar cualquier elemento de L_1 con cualquier elemento de L_2; específicamente,

$$L_1 L_2 = \{xy : x \in L_1, y \in L_2\}.$$

Operaciones sobre lenguajes II

- Definimos L^n como L concatenado consigo mismo n veces, con los casos especiales
$$L^0 = \{\lambda\} \text{ y } L^1 = L,$$
para todo lenguaje L.

Operaciones sobre lenguajes II

- Finalmente, definimos la **cerradura estrella** de un lenguaje como

$$L^* = \bigcup_{i \geq 0} L^i$$

- Finalmente, definimos la **cerradura estrella** de un lenguaje como

$$L^* = \bigcup_{i \geq 0} L^i$$

- y la **cerradura positiva** como

$$L^+ = \bigcup_{i \geq 1} L^i$$

Ejemplo 3

Si

$$L = \{a^n b^n : n \geq 0\},$$

entonces

$$L^2 = \{a^n b^n a^m b^m : n \geq 0, m \geq 0\}.$$

Ejemplo 3

Si

$$L = \{a^n b^n : n \geq 0\},$$

entonces

$$L^2 = \{a^n b^n a^m b^m : n \geq 0, m \geq 0\}.$$

Tenga en cuenta que n y m no están relacionados; la cadena aabbaaabbb está en L_2.

Ejemplo 3

Ejemplo 3

Si

$$L = \{a^n b^n : n \geq 0\},$$

entonces

$$L^2 = \{a^n b^n a^m b^m : n \geq 0, m \geq 0\}.$$

Tenga en cuenta que n y m no están relacionados; la cadena aabbaaabbb está en L_2. La reversa de L se describe fácilmente en notación de conjuntos como

$$L^R = \{b^n a^n : n \geq 0\},$$

Ejemplo 3

pero es considerablemente más difícil describir $\overline{L}$ o L^* de esta manera.

Ejemplo 3

Unos pocos intentos lo convencerán rápidamente de la limitación de la notación de conjuntos para la especificación de lenguajes complicados. $\square$

Motivación Gramáticas

- Para estudiar lenguajes matemáticamente, necesitamos un mecanismo para describirlos.

Motivación Gramáticas

- El lenguaje cotidiano es impreciso y ambiguo, por lo que las descripciones informales en lenguaje natural a menudo son inadecuadas.

Motivación Gramáticas

- La notación de conjuntos utilizada en los Ejemplos 2 y 3 es más adecuada, pero limitada.

- A medida que avancemos, aprenderemos sobre varios mecanismos de definición de lenguajes que son útiles en diferentes circunstancias.

Motivación Gramáticas

- Aquí presentamos uno común y poderoso, la noción de **gramática**

Gramáticas I

- Una gramática para el idioma inglés nos dice si una oración en particular está bien formada o no.

- Una gramática para el idioma inglés nos dice si una oración en particular está bien formada o no.
- Una regla típica de la gramática inglesa es "una oración puede constar de una frase nominal seguida de un predicado". Más concisamente escribimos esto como

$$< sentence > \to < noun\_phrase >< predicate >,$$

con la interpretación obvia.

Gramáticas I

- Una gramática para el idioma inglés nos dice si una oración en particular está bien formada o no.
- Una regla típica de la gramática inglesa es “una oración puede constar de una frase nominal seguida de un predicado”. Más concisamente escribimos esto como

$$< sentence > \to < noun\_phrase >< predicate >,$$

  con la interpretación obvia.
- Por supuesto, esto no es suficiente para tratar con oraciones reales.

Gramáticas I

- Una gramática para el idioma inglés nos dice si una oración en particular está bien formada o no.
- Una regla típica de la gramática inglesa es "una oración puede constar de una frase nominal seguida de un predicado". Más concisamente escribimos esto como

$$< sentence > \to < noun\_phrase >< predicate >,$$

  con la interpretación obvia.
- Por supuesto, esto no es suficiente para tratar con oraciones reales.
- Ahora debemos proporcionar definiciones para los constructos recién introducidos $< noun\_phrase >$ y $< predicate >$.

Gramáticas II

- Si lo hacemos mediante

$$< noun\_phrase > \rightarrow < article >< noun >,$$
$$< predicate > \rightarrow < verb >$$

Gramáticas II

- Si lo hacemos mediante

$$< noun\_phrase > \rightarrow < article >< noun >,$$
$$< predicate > \rightarrow < verb >$$

- y si asociamos las palabras "a" y "the" con $< article >$, "boy" y "dog" con $< noun >$, y "runs" y "walks" con $< verb >$,

- Si lo hacemos mediante

$$< noun\_phrase > \rightarrow < article >< noun >,$$
$$< predicate > \rightarrow < verb >$$

- y si asociamos las palabras "a" y "the" con $< article >$, "boy" y "dog" con $< noun >$, y "runs" y "walks" con $< verb >$,
- entonces la gramática nos dice que las oraciones "a boy runs" y "the dog walks" están correctamente formadas.

- Si lo hacemos mediante

$$< noun\_phrase > \to < article >< noun >,$$
$$< predicate > \to < verb >$$

- y si asociamos las palabras “a” y “the” con $< article >$, “boy” y “dog” con $< noun >$, y “runs” y “walks” con $< verb >$,
- entonces la gramática nos dice que las oraciones “a boy runs” y “the dog walks” están correctamente formadas.
- Si tuviéramos que dar una gramática completa, entonces, en teoría, cada oración bien formada podría explicarse de esta manera.

Gramáticas III

- Este ejemplo ilustra la definición de un concepto general en términos de conceptos simples.

Gramáticas III

- Comenzamos con el concepto de nivel superior, aquí $< sentence >$, y lo reducimos sucesivamente a los bloques de construcción irreductibles del lenguaje.

Gramáticas III

- La generalización de estas ideas nos lleva a gramáticas formales.

Definición 1

Una gramática G se define como un cuadruple

$$G = (V, T, S, P),$$

donde

Definición 1

Una gramática G se define como un cuadruple

$$G = (V, T, S, P),$$

donde

- V es un conjunto finito de objetos llamados **variables**,

Definición formal de Gramática

Definición 1

Una gramática G se define como un cuadruple

$$G = (V, T, S, P),$$

donde

- V es un conjunto finito de objetos llamados **variables**,
- T es un conjunto finito de objetos llamados **símbolos terminales**,

Definición formal de Gramática

Definición 1

Una gramática G se define como un cuadruple

$$G = (V, T, S, P),$$

donde

- V es un conjunto finito de objetos llamados **variables**,
- T es un conjunto finito de objetos llamados **símbolos terminales**,
- $S \in V$ es un símbolo especial llamado variable **inicial**,

Definición 1

Una gramática G se define como un cuadruple

$$G = (V, T, S, P),$$

donde

- V es un conjunto finito de objetos llamados **variables**,
- T es un conjunto finito de objetos llamados **símbolos terminales**,
- $S \in V$ es un símbolo especial llamado variable **inicial**,
- P es un conjunto finito de **producciones**.

Definición 1

Una gramática G se define como un cuadruple

$$G = (V, T, S, P),$$

donde

- V es un conjunto finito de objetos llamados **variables**,
- T es un conjunto finito de objetos llamados **símbolos terminales**,
- $S \in V$ es un símbolo especial llamado variable **inicial**,
- P es un conjunto finito de **producciones**.

Se supondrá, sin más mención, que los conjuntos V y T no están vacíos y que son disjuntos.

Derivación I

- Las reglas de producción son el corazón de una gramática; especifican cómo la gramática transforma una cadena en otra, y a través de esto definen un lenguaje asociado a la gramática.

Derivación I

- En nuestra discusión asumiremos que todas las reglas de producción son de la forma
$$x \to y,$$
donde x es un elemento de $(V \cup T)^+$ y y está en $(V \cup T)^*$.

Derivación II

- Las producciones se aplican de la siguiente manera: Dada una cadena w de la forma
$$w = uxv,$$
decimos que la producción $x \to y$ es aplicable a esta cadena, y podemos usarla para reemplazar x con y, obteniendo así una nueva cadena
$$z = uyv.$$

- Las producciones se aplican de la siguiente manera: Dada una cadena w de la forma
$$w = uxv,$$
decimos que la producción $x \to y$ es aplicable a esta cadena, y podemos usarla para reemplazar x con y, obteniendo así una nueva cadena
$$z = uyv.$$
- Esto se escribe como
$$w \Rightarrow z.$$
Decimos que w **deriva** z o que z se deriva de w.

Derivación II

- Las producciones se aplican de la siguiente manera: Dada una cadena w de la forma
$$w = uxv,$$
decimos que la producción $x \to y$ es aplicable a esta cadena, y podemos usarla para reemplazar x con y, obteniendo así una nueva cadena
$$z = uyv.$$
- Esto se escribe como
$$w \Rightarrow z.$$
Decimos que w **deriva** z o que z se deriva de w.
- Las cadenas sucesivas se derivan aplicando las producciones de la gramática en orden arbitrario.

Derivación III

- Una producción se puede utilizar siempre que sea aplicable y se puede aplicar tantas veces como se desee. Si

$$w_1 \Rightarrow w_2 \Rightarrow \cdots \Rightarrow w_n,$$

decimos que w_1 deriva w_n y lo denotamos mediante

$$w_1 \stackrel{*}{\Rightarrow} w_n.$$

Derivación III

- Una producción se puede utilizar siempre que sea aplicable y se puede aplicar tantas veces como se desee. Si

$$w_1 \Rightarrow w_2 \Rightarrow \cdots \Rightarrow w_n,$$

decimos que w_1 deriva w_n y lo denotamos mediante

$$w_1 \overset{*}{\Rightarrow} w_n.$$

- El $*$ indica que se puede tomar un número no especificado de pasos (incluido cero) para derivar w_n partiendo de w_1.

Derivación III

- Una producción se puede utilizar siempre que sea aplicable y se puede aplicar tantas veces como se desee. Si

$$w_1 \Rightarrow w_2 \Rightarrow \cdots \Rightarrow w_n,$$

decimos que w_1 deriva w_n y lo denotamos mediante

$$w_1 \stackrel{*}{\Rightarrow} w_n.$$

- El $*$ indica que se puede tomar un número no especificado de pasos (incluido cero) para derivar w_n partiendo de w_1.
- Al aplicar las reglas de producción en un orden diferente, una gramática determinada normalmente puede generar muchas cadenas.

Derivación III

- Una producción se puede utilizar siempre que sea aplicable y se puede aplicar tantas veces como se desee. Si

$$w_1 \Rightarrow w_2 \Rightarrow \cdots \Rightarrow w_n,$$

decimos que w_1 deriva w_n y lo denotamos mediante

$$w_1 \stackrel{*}{\Rightarrow} w_n.$$

- El $*$ indica que se puede tomar un número no especificado de pasos (incluido cero) para derivar w_n partiendo de w_1.
- Al aplicar las reglas de producción en un orden diferente, una gramática determinada normalmente puede generar muchas cadenas.
- El conjunto de todas esas cadenas terminales es el lenguaje definido o generado por la gramática.

Definición formal de $L(G)$

Definición 2

Sea $G = (V, T, S, P)$ una gramática. Entonces el conjunto

$$L(G) = \left\{ w \in T^* : S \stackrel{*}{\Rightarrow} w \right\}$$

es el lenguaje generado por G.

Definición 2

Sea $G = (V, T, S, P)$ una gramática. Entonces el conjunto

$$L(G) = \left\{ w \in T^* : S \overset{*}{\Rightarrow} w \right\}$$

es el lenguaje generado por G.

Si $w \in L(G)$, entonces la secuencia

$$S \Rightarrow w_1 \Rightarrow w_2 \Rightarrow \cdots \Rightarrow w_n \Rightarrow w$$

es una derivación de la oración w.

Definición formal de $L(G)$

Definición 2

Sea $G = (V, T, S, P)$ una gramática. Entonces el conjunto

$$L(G) = \left\{ w \in T^* : S \overset{*}{\Rightarrow} w \right\}$$

es el lenguaje generado por G.

Si $w \in L(G)$, entonces la secuencia

$$S \Rightarrow w_1 \Rightarrow w_2 \Rightarrow \cdots \Rightarrow w_n \Rightarrow w$$

es una derivación de la oración w.

Las cadenas $S, w_1, w_2, \ldots, w_n$, que contienen tanto variables como terminales, se denominan formas oracionales de la derivación.

Ejemplo 4 I

Ejemplo 4

Considere la gramática

$$G = (\{S\}, \{a, b\}, S, P)$$

con P dado por

$$\begin{aligned} S &\to aSb, \\ S &\to \lambda. \end{aligned}$$

Ejemplo 4 I

Ejemplo 4

Considere la gramática

$$G = (\{S\}, \{a, b\}, S, P)$$

con P dado por

$$\begin{aligned} S &\to aSb, \\ S &\to \lambda. \end{aligned}$$

$$S \Rightarrow aSb \Rightarrow aaSbb \Rightarrow aabb,$$

Ejemplo 4 I

Ejemplo 4

Considere la gramática

$$G = (\{S\}, \{a, b\}, S, P)$$

con P dado por

$$S \to aSb,$$
$$S \to \lambda.$$

$$S \Rightarrow aSb \Rightarrow aaSbb \Rightarrow aabb,$$

así que podemos escribir

$$S \overset{*}{\Rightarrow} aabb.$$

Ejemplo 4 I

Ejemplo 4

Considere la gramática

$$G = (\{S\}, \{a, b\}, S, P)$$

con P dado por

$$S \to aSb,$$
$$S \to \lambda.$$

$$S \Rightarrow aSb \Rightarrow aaSbb \Rightarrow aabb,$$

así que podemos escribir

$$S \stackrel{*}{\Rightarrow} aabb.$$

La cadena $aabb$ es una oración en el lenguaje generado por G, mientras que $aaSbb$ es una forma oracional.

Ejemplo 4 II

- Una gramática G define completamente a $L(G)$, pero puede que no sea fácil obtener una descripción muy explícita del lenguaje a partir de la gramática.

Ejemplo 4 II

- Aquí, sin embargo, la respuesta es bastante clara.

Ejemplo 4 II

- No es difícil conjeturar que

$$L(G) = \{a^n b^n; n \geq 0\},$$

y es fácil probarlo.

Ejemplo 4 II

- Si notamos que la regla $S \to aSb$ es recursiva, se sugiere inmediatamente una demostración por inducción.

- Primero mostramos que todas las formas oracionales deben tener la forma

$$w_i = a^i S b^i. \tag{2}$$

Ejemplo 4 III

- Primero mostramos que todas las formas oracionales deben tener la forma
$$w_i = a^i S b^i. \tag{2}$$
- Suponga que (2) se cumple para todas las formas oracionales w_i de longitud $2i + 1$ o menos. Para obtener otra forma oracional (que no es una oración), solo podemos aplicar la producción $S \to aSb$. Esto nos da
$$a^i S b^i \Rightarrow a^{i+1} S b^{i+1},$$
de modo que toda forma oracional de longitud $2i + 3$ también tiene la forma (2).

Ejemplo 4 IV

- Dado que (2) es obviamente cierto para $i = 1$, se cumple por inducción para todo i.

Ejemplo 4 IV

- Dado que (2) es obviamente cierto para $i = 1$, se cumple por inducción para todo i.
- Finalmente, para obtener una oración, debemos aplicar la producción $S \to \lambda$, y vemos que

$$S \stackrel{*}{\Rightarrow} a^n S b^n \Rightarrow a^n b^n$$

representa todas las posibles derivaciones.

Ejemplo 4 IV

- Dado que (2) es obviamente cierto para $i = 1$, se cumple por inducción para todo i.
- Finalmente, para obtener una oración, debemos aplicar la producción $S \to \lambda$, y vemos que

$$S \overset{*}{\Rightarrow} a^n S b^n \Rightarrow a^n b^n$$

representa todas las posibles derivaciones.

- Por tanto, G sólo puede derivar cadenas de la forma $a^n b^n$.

Ejemplo 4 IV

- Dado que (2) es obviamente cierto para $i = 1$, se cumple por inducción para todo i.
- Finalmente, para obtener una oración, debemos aplicar la producción $S \to \lambda$, y vemos que

$$S \stackrel{*}{\Rightarrow} a^n S b^n \Rightarrow a^n b^n$$

  representa todas las posibles derivaciones.
- Por tanto, G sólo puede derivar cadenas de la forma $a^n b^n$.
- También tenemos que demostrar que se pueden derivar todas las cadenas de esta forma.

Ejemplo 4 IV

- Dado que (2) es obviamente cierto para $i = 1$, se cumple por inducción para todo i.
- Finalmente, para obtener una oración, debemos aplicar la producción $S \to \lambda$, y vemos que

$$S \stackrel{*}{\Rightarrow} a^n S b^n \Rightarrow a^n b^n$$

  representa todas las posibles derivaciones.
- Por tanto, G sólo puede derivar cadenas de la forma $a^n b^n$.
- También tenemos que demostrar que se pueden derivar todas las cadenas de esta forma.
- Esto es fácil; simplemente aplicamos $S \to aSb$ tantas veces como sea necesario, seguido de $S \to \lambda$. $\square$

Ejemplo 5 I

Ejemplo 5

Encuentra una gramática que genere

$$L = \{a^n b^{n+1} : n \geq 0\}.$$

Ejemplo 5 I

Ejemplo 5

Encuentra una gramática que genere

$$L = \{a^n b^{n+1} : n \geq 0\}.$$

- La idea detrás del ejemplo anterior se puede extender a este caso.

Ejemplo 5 I

Ejemplo 5

Encuentra una gramática que genere

$$L = \{a^n b^{n+1} : n \geq 0\}.$$

- La idea detrás del ejemplo anterior se puede extender a este caso.
- Todo lo que tenemos que hacer es generar una b extra.

Ejemplo 5 I

Ejemplo 5

Encuentra una gramática que genere

$$L = \{a^n b^{n+1} : n \geq 0\}.$$

- La idea detrás del ejemplo anterior se puede extender a este caso.
- Todo lo que tenemos que hacer es generar una b extra.
- Esto se puede hacer con una producción $S \to Ab$ y con otras producciones elegidas para que A pueda derivar el lenguaje del ejemplo anterior.

Ejemplo 5 II

- Razonando de esta manera, obtenemos la gramática $G = (\{S, A\}, \{a, b\}, S, P)$, con producciones

$$
\begin{aligned}
S &\rightarrow Ab, \\
A &\rightarrow aAb, \\
A &\rightarrow \lambda.
\end{aligned}
$$

Ejemplo 5 II

- Razonando de esta manera, obtenemos la gramática $G = (\{S, A\}, \{a, b\}, S, P)$, con producciones

$$S \to Ab,$$
$$A \to aAb,$$
$$A \to \lambda.$$

- Derive algunas oraciones específicas para convencerse de que esto funciona. $\square$

Advertencia

- Los ejemplos anteriores son bastante fáciles, por lo que argumentos rigurosos pueden parecer superfluos.

Advertencia

- Pero a menudo no es tan fácil encontrar una gramática para un lenguaje descrito de manera informal o dar una caracterización intuitiva del lenguaje definido por una gramática.

Advertencia

- Para mostrar que un lenguaje dado es generado por una determinada gramática G, debemos ser capaces de demostrar que:

Advertencia

- Para mostrar que un lenguaje dado es generado por una determinada gramática G, debemos ser capaces de demostrar que:
  1. cada $w \in L$ puede derivarse de S usando G y

Advertencia

- Para mostrar que un lenguaje dado es generado por una determinada gramática G, debemos ser capaces de demostrar que:
  1. cada $w \in L$ puede derivarse de S usando G y
  2. cada cadena así derivada está en L.

Ejemplo 6 I

Ejemplo 6

Tome $\Sigma = \{a, b\}$, y sean $n_a(w)$ y $n_b(w)$ el número de símbolos a y b en la cadena w, respectivamente. Luego la gramática G con producciones

$$S \to SS,$$
$$S \to \lambda,$$
$$S \to aSb,$$
$$S \to bSa$$

genera el lenguage

$$L = \{w : n_a(w) = n_b(w)\}.$$

Ejemplo 6 II

- Esta afirmación no es tan obvia y debemos proporcionar argumentos convincentes.

Ejemplo 6 II

- En primer lugar, está claro que cada forma oracional de G tiene el mismo número de símbolos a y b, ya que las únicas producciones que generan una a, a saber, $S \to aSb$ y $S \to bSa$, generan simultáneamente una b.

Ejemplo 6 II

- Por lo tanto, cada elemento de $L(G)$ está en L.

Ejemplo 6 II

- Es un poco más difícil ver que cada cadena en L se puede derivar con G.

Ejemplo 6 III

- Comencemos por analizar el problema en resumen, considerando las diversas formas que puede tener $w \in L$.

Ejemplo 6 III

- Suponga que w comienza con a y termina con b.

Ejemplo 6 III

- Entonces tiene la forma
$$w = aw_1b,$$
donde w_1 también está en L.

Ejemplo 6 III

- Podemos pensar en este caso como que es derivado a partir de

$$S \Rightarrow aSb$$

si S de hecho deriva cualquier cadena en L.

Ejemplo 6 IV

- Se puede hacer un argumento similar si w comienza con b y termina con a.

Ejemplo 6 IV

- Pero esto no se ocupa de todos los casos, ya que una cadena en L puede comenzar y terminar con el mismo símbolo.

Ejemplo 6 IV

- Si escribimos una cadena de este tipo, digamos $aabbba$, vemos que se puede considerar como la concatenación de dos cadenas más cortas $aabb$ y ba, ambas en L.

Ejemplo 6 IV

- ¿Es esto cierto en general?

Ejemplo 6 IV

- Para mostrar que esto es así, podemos usar el siguiente argumento: Supongamos que, comenzando en el extremo izquierdo de la cadena, contamos $+1$ para una a y -1 para una b.

Ejemplo 6 V

- Si una cadena w comienza y termina con a, entonces la cuenta será $+1$ después del símbolo más a la izquierda y -1 inmediatamente antes del más a la derecha.

Ejemplo 6 V

- Por lo tanto, la cuenta debe pasar por cero en algún lugar en el medio de la cadena, lo que indica que dicha cadena debe tener la forma

$$w = w_1 w_2$$

donde w_1 y w_2 están en L.

Ejemplo 6 V

- Este caso se puede resolver con la producción $S \to SS$.

Ejemplo 6 V

- Una vez que vemos el argumento de manera intuitiva, estamos listos para proceder con más rigor. Nuevamente usamos inducción.

Ejemplo 6 V

- Suponga que todo $w \in L$ con $|w| \leq 2n$ se puede derivar con G.

Ejemplo 6 V

- Tome cualquier $w \in L$ de longitud $2n+2$. Si $w = aw_1b$, entonces w_1 está en L y $|w_1| = 2n$. Por lo tanto, por suposición,

$$S \overset{*}{\Rightarrow} w_1$$

Ejemplo 6 VI

- Entonces

$$S \Rightarrow aSb \stackrel{*}{\Rightarrow} aw_1b = w$$

es posible, y w se puede derivar con G.

Ejemplo 6 VI

- Obviamente, se pueden hacer argumentos similares si $w = bw_1a$.

Ejemplo 6 VI

- Si w no tiene esta forma, es decir, si comienza y termina con el mismo símbolo, entonces el argumento de conteo nos dice que debe tener la forma $w = w_1 w_2$, con w_1 y w_2 ambos en L y de longitud menor que o igual a $2n$.

Ejemplo 6 VI

- Por lo tanto, nuevamente vemos que

$$S \Rightarrow SS \stackrel{*}{\Rightarrow} w_1 S \stackrel{*}{\Rightarrow} w_1 w_2 = w$$

  es posible.

Ejemplo 6 VI

- Ya que la hipótesis inductiva se satisface claramente para $n = 1$, se cumple el caso base y la afirmación es verdadera para todo n, completando la demostración. $\square$

Gramáticas equivalentes

- Normalmente, un lenguaje dado tiene muchas gramáticas que lo generan.

Gramáticas equivalentes

- Aunque estas gramáticas son diferentes, son equivalentes en cierto sentido.

Gramáticas equivalentes

- Decimos que dos gramáticas G_1 y G_2 son equivalentes si generan el mismo lenguaje, es decir, si

$$L(G_1) = L(G_2).$$

Gramáticas equivalentes

- Como veremos más adelante, no siempre es fácil ver si dos gramáticas son equivalentes.

Ejemplo 7

Considere la gramática $G_1 = (\{A, S\}, \{a, b\}, S, P_1)$, con P_1 que consta de las producciones

$$S \to aAb|\lambda,$$
$$A \to aAb|\lambda.$$

Ejemplo 7

Ejemplo 7

Considere la gramática $G_1 = (\{A, S\}, \{a, b\}, S, P_1)$, con P_1 que consta de las producciones

$$S \to aAb|\lambda,$$
$$A \to aAb|\lambda.$$

- Aquí presentamos una notación abreviada conveniente en la que varias reglas de producción con los mismos lados izquierdos se escriben en la misma línea, con los lados derechos alternativos separados por $|$.

Ejemplo 7

Considere la gramática $G_1 = (\{A, S\}, \{a, b\}, S, P_1)$, con P_1 que consta de las producciones

$$S \to aAb|\lambda,$$
$$A \to aAb|\lambda.$$

- En esta notación $S \to aAb|\lambda$ representa las dos producciones $S \to aAb$ y $S \to \lambda$.

Ejemplo 7

Considere la gramática $G_1 = (\{A, S\}, \{a, b\}, S, P_1)$, con P_1 que consta de las producciones

$$S \to aAb|\lambda,$$
$$A \to aAb|\lambda.$$

- Esta gramática es equivalente a la gramática G del ejemplo 4. La equivalencia es fácil de probar mostrando que

$$L(G_1) = \{a^n b^n : n \geq 0\}.$$

$\square$

- Un autómata es un modelo abstracto de una computadora digital.

Autómatas I

- Como tal, cada autómata incluye algunas características esenciales.

- Tiene un mecanismo de lectura de entrada.

- Se asumirá que la entrada es una cadena sobre un alfabeto dado, escrita en un archivo de entrada, que el autómata puede leer pero no cambiar.

- El archivo de entrada se divide en celdas, cada una de las cuales puede contener un símbolo.

- El mecanismo de entrada puede leer el archivo de entrada de izquierda a derecha, un símbolo a la vez.

- El mecanismo de entrada también puede detectar el final de la cadena de entrada (detectando una condición de final de archivo).

- El autómata puede producir una salida de alguna forma.

- Puede tener un dispositivo de **almacenamiento** temporal, que consta de un número ilimitado de celdas, cada una de las cuales puede contener un solo símbolo de un alfabeto (no necesariamente el mismo que el alfabeto de entrada).

- El autómata puede leer y cambiar el contenido de las celdas de almacenamiento.

- Finalmente, el autómata tiene una **unidad de control**, que puede estar en cualquiera de un número finito de **estados internos**, y que puede cambiar de estado de alguna manera definida.

- La figura 1 muestra una representación esquemática de un autómata general.

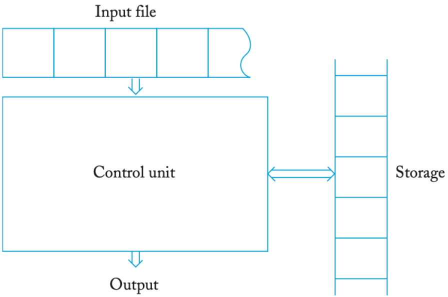

Figure 1: Representación esquemática de un autómata general.

Autómatas IV

- Se supone que un autómata opera en un marco de tiempo discreto.

- En cualquier momento dado, la unidad de control se encuentra en algún estado interno y el mecanismo de entrada está escaneando un símbolo particular en el archivo de entrada.

Autómatas IV

- El estado interno de la unidad de control en el siguiente paso de tiempo lo determina la **función de transición** o **estado siguiente**.

- Esta función de transición proporciona el siguiente estado en términos del estado actual, el símbolo de entrada actual y la información que se encuentra actualmente en el almacenamiento temporal.

- Durante la transición de un intervalo de tiempo al siguiente, se puede producir una salida o cambiar la información del almacenamiento temporal.

Autómatas V

- El término **configuración** se utilizará para referirse a un estado particular de la unidad de control, archivo de entrada y almacenamiento temporal.

Autómatas V

- La transición del autómata de una configuración a la siguiente se denominará **movimiento**.

Autómatas V

- Este modelo general cubre todos los autómatas que discutiremos en este curso.

Autómatas VI

- Un control de estado finito será común a todos los casos específicos, pero las diferencias surgirán de la forma en que se puede producir la salida y la naturaleza del almacenamiento temporal.

Autómatas VI

- Como veremos, la naturaleza del almacenamiento temporal gobierna el poder de diferentes tipos de autómatas.

Autómatas VI

- Para discusiones posteriores, será necesario distinguir entre autómatas deterministas y autómatas no deterministas.

- Un autómata determinista es aquel en el que cada movimiento está determinado de forma única por la configuración actual.

Autómatas VI

- Si conocemos el estado interno, la entrada y el contenido del almacenamiento temporal, podemos predecir exactamente el comportamiento futuro del autómata.

- En un autómata no determinista, esto no es así. En cada punto, un autómata no determinista puede tener varios movimientos posibles, por lo que solo podemos predecir un conjunto de acciones posibles.

Autómatas VII

- La relación entre los autómatas deterministas y no deterministas de varios tipos jugará un papel importante en nuestro estudio.

Autómatas VII

- Un autómata cuya respuesta de salida se limita a un simple "sí" o "no" se denomina **aceptador**.

Autómatas VII

- Presentado con una cadena de entrada, un aceptador acepta la cadena o la rechaza.

Autómatas VII

- Un autómata más general, capaz de producir cadenas de símbolos como salida, se llama **transductor**.

Ejercicios I

1. ¿Cuántas subcadenas aab hay en ww^Rw, donde $w = aabbab$?
2. Use inducción sobre n para demostrar que $|u^n| = n|u|$ para todas las cadenas u y para todo n.
3. El reverso de una cadena, introducido informalmente anteriormente, se puede definir con mayor precisión mediante las reglas recursivas

$$a^R = a,$$
$$(wa)^R = aw^R,$$

para todo $a \in \Sigma$, $w \in \Sigma^*$. Use esto para demostrar que

$$(uv)^R = v^R u^R,$$

para todo $u, v \in \Sigma^+$.

Ejercicios II

4. Sea $L = \{ab, aa, baa\}$. ¿Cuáles de las siguientes cadenas están en $L^* : abaabaaabaa, aaaabaaaa, baaaaabaaaab, baaaabaa$? ¿Qué cadenas están en L^4?
5. Sea $\Sigma = \{a, b\}$ y $L = \{aa, bb\}$. Utilice la notación de conjuntos para describir $\overline{L}$.
6. Demuestre que
$$(L_1 L_2)^R = L_2^R L_1^R$$
para todos los lenguajes L_1 y L_2.
7. Encuentre una gramática para el lenguaje $L = \{a^n,$ donde n es par$\}$.
8. Dé una descripción sencilla del lenguaje generado por la gramática con producciones
$$S \to aaA,$$
$$A \to bS,$$
$$S \to \lambda.$$

Ejercicios III

9. Encuentre tres cadenas en el lenguaje generado por

$$S \to aSb|bSa|a.$$

10. Complete los argumentos en el Ejemplo 7, mostrando que $L(G_1)$ en efecto genera el lenguaje dado en el Ejemplo 4.