Generalidades sobre lenguajes.

U.D. Computación

DSIC - UPV

September 28, 2018

Definiciones básicas: Alfabeto

- *Alfabeto*: Conjunto finito de símbolos
  - $\Sigma = \{a, b, c\}$
  - $\Gamma = \{0, 1\}$
  - $\Delta_1 = \{\triangle, \square, \bigcirc\}$
  - $\Delta_2 = \{N, S, E, W\}$
- No son alfabetos:
  - $\emptyset$
  - $\mathbb{N}$

Definiciones básicas: Palabra

- (También *frase* o *cadena*) secuencia finita y ordenada de símbolos de un alfabeto
  - Sobre $\{a, b\}$: $x = aaba$, $y = aa$
  - Sobre $\{0, 1, 2\}$: $x = 2110$, $y = 0101$
- *palabra vacía*: λ.

Definiciones básicas: Longitud

- *Longitud* de una palabra: número de símbolos que contiene.
  Siendo x, y palabras sobre Σ, $a \in \Sigma$, se define:

$$|x| = \begin{cases} 0 & \text{si } x = \lambda \\ 1 + |y| & \text{si } x = ay \end{cases}$$

- $|x|_a$ es el número de veces que aparece el símbolo a en x.

Definiciones básicas: Palabras sobre Σ

- Σ^n conjunto de palabras de longitud n sobre el alfabeto
- $\Sigma^* = \bigcup_{i \geq 0} \Sigma^i$
- $\Sigma^+ = \bigcup_{i > 0} \Sigma^i$

Definiciones básicas: Orden canónico

- El orden alfabético ($<_\Sigma$) no permite una enumeración efectiva de las palabras sobre un alfabeto Σ
- Dadas dos palabras x, y sobre un alfabeto Σ, se define *orden canónico* como:

$$x < y \text{ si } \begin{cases} |x| < |y| \\ (|x| = |y|) \ \wedge \ (x = uav, \ y = ubw, \ a <_\Sigma b) \end{cases}$$

Operaciones sobre palabras: Concatenación

Dadas $x = a_1 a_2 \cdots a_m$, $y = b_1 b_2 \cdots b_n$, $a_i, b_j \in \Sigma$, se define *concatenación* de x e y como:

$$x \cdot y = xy = a_1 a_2 \cdots a_m b_1 b_2 \cdots b_n$$

Considerando la concatenación, se define *potencia* de una palabra como:

$$x^n = \begin{cases} \lambda & \text{si } n = 0 \\ x \cdot x^{n-1} = x^{n-1} \cdot x & \text{si } n > 0 \end{cases}$$

Operaciones sobre palabras: Concatenación

Propiedades de la concatenación

Sean $x, y, z \in \Sigma^*$ y $a \in \Sigma$

1. Asociativa: $(x \cdot y) \cdot z = x \cdot (y \cdot z)$.
2. Existencia de *elemento neutro* (λ): $x\lambda = \lambda x = x$.
3. $|xy| = |x| + |y|$

Operaciones sobre palabras: Segmento, prefijo, sufijo

Dados x y t, palabras de Σ^*

- t es un *segmento* de x si existen u y v tales que $x = u \cdot t \cdot v$.
- Si $u = \lambda$, entonces t es un *prefijo* de x.
- Si $v = \lambda$, entonces t es un *sufijo* de x.

Operaciones sobre palabras: Reverso

Dados $x, y \in \Sigma^*$ y un símbolo a del alfabeto, se define el *reverso* de una palabra como:

$$\begin{cases} \lambda^r = \lambda \\ a^r = a \\ (ax)^r = x^r a \\ (xa)^r = a x^r \end{cases}$$

Operaciones sobre palabras: Reverso

Propiedades del reverso

Sean x, y dos palabras en Σ^*

1. $(x^r)^r = x$
2. $(xy)^r = y^r x^r$
3. $(x^n)^r = (x^r)^n$ para cualquier entero $n \geq 0$

Lenguajes: Definiciones

Un *lenguaje L* es un subconjunto de Σ^*

se incluyen:

- $\emptyset$ (lenguaje vacío, no contiene ninguna palabra)
- Σ^* (todas las palabras posibles sobre Σ)

- Un lenguaje se denomina finito si tiene un número finito de palabras
- Caso contrario es infinito numerable

Lenguajes: Operaciones booleanas

- Unión: $L_1 \cup L_2 = \{x \in \Sigma^* \ : \ x \in L_1 \vee x \in L_2\}$
- Intersección: $L_1 \cap L_2 = \{x \in \Sigma^* \ : \ x \in L_1 \wedge x \in L_2\}$
- Complementación: $\overline{L} = \{x \in \Sigma^* \ : \ x \notin L\}$

Lenguajes: Operaciones booleanas

Propiedades Unión e Intersección

- Asociativa
- Conmutativa
- Elemento neutro $(\emptyset, \Sigma^*)$
  - Unión: $\emptyset$
  - Intersección: Σ^*
- Distributivas:
  - $L_1 \cup (L_2 \cap L_3) = (L_1 \cup L_2) \cap (L_1 \cup L_3)$
  - $L_1 \cap (L_2 \cup L_3) = (L_1 \cap L_2) \cup (L_1 \cap L_3)$

Lenguajes: Operaciones booleanas

Propiedades Complementación

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

Lenguajes: Operaciones booleanas

Operaciones definidas a partir de las booleanas

- Diferencia: $L_1 - L_2 = L_1 \cap \overline{L}_2$.
- Diferencia simétrica: $L_1 \ominus L_2 = \left(L_1 \cap \overline{L}_2\right) \cup \left(\overline{L}_1 \cap L_2\right)$.

Lenguajes: Operaciones racionales. Producto

$$L_1 \cdot L_2 = \{xy \in \Sigma^* \ : \ x \in L_1 \wedge y \in L_2\}$$

Propiedades

- (No conmutativa). $L_1 \cdot L_2$ no necesariamente igual a $L_2 \cdot L_1$
- (Asociativa) $(L_1 \cdot L_2) \cdot L_3 = L_1 \cdot (L_2 \cdot L_3)$
- (Elemento neutro) $L \cdot \{\lambda\} = \{\lambda\} \cdot L = L$
- (Anulador) $L \cdot \emptyset = \emptyset \cdot L = \emptyset$
- $\lambda \in L_1 \cdot L_2 \Leftrightarrow \lambda \in L_1 \wedge \lambda \in L_2$
- $L_1 \cdot (L_2 \cup L_3) = L_1 \cdot L_2 \cup L_1 \cdot L_3$
- $L_1 \cdot (L_2 \cap L_3) \subseteq L_1 \cdot L_2 \cap L_1 \cdot L_3$
  - Ejemplo: $L_1 = \{a, ab\}$, $L_2 = \{a\}$, $L_3 = \{ba\}$.

Lenguajes: Operaciones racionales. Potencia

$$L^n = \begin{cases} \{\lambda\} & \text{si } n = 0 \\ LL^{n-1} = L^{n-1}L & \text{si } n > 0 \end{cases}$$

Ejemplos

- $\{aa, b\}^2 = \{bb, aab, baa, aaaa\}$
- $\emptyset^0 = (\Sigma^*)^0 = \{\lambda\}^0 = \{\lambda\}$

Lenguajes: Operaciones racionales. Cierre

Cierre estrella

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

Cierre positivo

$$L^+ = \bigcup_{i > 0} L^i$$

Lenguajes: Operaciones racionales. Cierre

Relación entre ambos

$$L^+ = \begin{cases} L^* & \text{si } \lambda \in L \\ L^* - \{\lambda\} & \text{si } \lambda \notin L \end{cases}$$

Lenguajes: Operaciones racionales. Cierre

Propiedades Cierre estrella y positivo

1. $L \subseteq L^+ \subseteq L^*$ (Puesto que $L = L^1$).
2. $L_1 \subseteq L_2 \Rightarrow L_1^n \subseteq L_2^n$ $(\forall n \in \mathbb{N})$.
3. $L_1 \subseteq L_2 \Rightarrow L_1^* \subseteq L_2^*$ $\left(L_1^+ \subseteq L_2^+\right)$
4. $(L^*)^* = L^*$
5. $(L^+)^+ = L^+$
6. $L^+ = L^*L = LL^*$
7. $(L^+)^* = L^*$
8. $(L^*)^+ = L^*$

Lenguajes: Cocientes

Cociente por la derecha

$$u^{-1}L = \{v \in \Sigma^* \ : \ uv \in L\}$$

Cociente por la izquierda

$$Lu^{-1} = \{v \in \Sigma^* \ : \ vu \in L\}$$

Lenguajes: Cocientes

Al cociente de un lenguaje respecto una palabra en ocasiones se le denomina *derivada*

Propiedades ($u, v \in \Sigma^*$, $a \in \Sigma$)

- $L_1 \subseteq L_2 \Rightarrow u^{-1}L_1 \subseteq u^{-1}L_2$
- $u^{-1}(L_1 \cup L_2) = u^{-1}L_1 \cup u^{-1}L_2$
- $u^{-1}(L_1 \cap L_2) = u^{-1}L_1 \cap u^{-1}L_2$
- $$a^{-1}(L_1 L_2) = \begin{cases} (a^{-1}L_1) L_2 & \text{si } \lambda \notin L_1 \\ (a^{-1}L_1) L_2 \cup a^{-1}L_2 & \text{si } \lambda \in L_1 \end{cases}$$
- $a^{-1}L^* = (a^{-1}L) L^*$
- $(uv)^{-1} L = v^{-1}(u^{-1}L)$

Lenguajes: Homomorfismos

Dados dos alfabetos Σ y Γ, un *homomorfismo* es una aplicación:

$$h : \Sigma \rightarrow \Gamma^*$$

Esta definición puede extenderse a palabras:

$$h : \Sigma^* \rightarrow \Gamma^*$$

$$\begin{cases} h(\lambda) = \lambda \\ h(xa) = h(x)h(a) \end{cases}$$

y a lenguajes:

$$h(L) = \{h(x) : x \in L\}$$

Lenguajes: Homomorfismo

Ejemplos

$$L_1 = \{\lambda, aa, bab, bbba\}$$

$$L_2 = \{x \in \{a, b\}^* \ : \ aa \notin Seg(x)\}$$

$$\begin{cases} h(a) = \lambda \\ h(b) = 1 \end{cases} \qquad \begin{cases} g(a) = 01 \\ g(b) = 1 \end{cases}$$

1. $h(L_1) = \{\lambda, 11, 111\}$
2. $h(L_2) = \{1\}^*$
3. $g(\{a, b\}^*) = \{x \in \{0, 1\}^* \ : \ 00 \notin Seg(x) \wedge 0 \notin Suf(x)\}$

Lenguajes: Homomorfismo inverso

Dado un homomorfismo $h : \Sigma^* \to \Gamma^*$, se define el homomorfismo inverso como:

$$h^{-1}(y) = \{x \in \Sigma^* \ : \ h(x) = y\}$$

La operación puede extenderse para considerar lenguajes:

$$h^{-1}(L) = \{x \in \Sigma^* : h(x) \in L\}$$

Lenguajes: Homomorfismo inverso

Ejemplos

$$L_1 = \{\lambda, aa, abab, bbba\}$$

$$L_2 = \{x \in \{a,b\}^* \ : \ aa \notin Seg(x)\}$$

$$\begin{cases} h(0) = ab \\ h(1) = ba \end{cases} \qquad \begin{cases} g(0) = aa \\ g(1) = bab \end{cases}$$

1. $h^{-1}(L_1) = \{\lambda, 00\}$
2. $g^{-1}(L_2) = \{x \in \{0,1\}^* \ : \ |x|_0 = 0\} = \{1\}^*$
3. $h^{-1}(L_2) = \{x \in \{0,1\}^* \ : \ 10 \notin Seg(x)\}$

Lenguajes: Otras operaciones. Reverso

Una forma de extender una operación definida sobre palabras a lenguajes es aplicar la operación a toda palabra del lenguaje:

$$L^r = \{x^r : x \in L\}$$

Lenguajes: Otras operaciones. Reverso

Propiedades

1. Si $\Sigma = \{a\}$, $L^r = L$.
2. $(L_1 L_2)^r = L_2^r L_1^r$
3. $(L^n)^r = (L^r)^n$
4. $(L^*)^r = (L^r)^*$
5. $(L^r)^r = L$

Lenguajes: Otras operaciones. Segmento, prefijo, sufijo

Cuando una operación definida sobre palabras devuelve un lenguaje, la extensión a lenguajes es diferente:

$$Seg(L) = \bigcup_{x \in L} Seg(x)$$

$$Pref(L) = \bigcup_{x \in L} Pref(x)$$

$$Suf(L) = \bigcup_{x \in L} Suf(x)$$

Clases de lenguajes

Una clase de lenguajes es una colección o conjunto no vacío de lenguajes.

Ejemplos

1. $\mathcal{L}_{FIN}$ Clase de los lenguajes finitos
2. $\mathcal{L}_{PAL} = \{L \subseteq \Sigma^* \;:\; x \in L \rightarrow x = x^r\}$
   (clase de los lenguajes palindrómicos)
3. $\mathcal{L}_{PAR} = \{L \subseteq \Sigma^* \;:\; x \in L \rightarrow |x| \mod 2 = 0\}$
   (clase de los lenguajes pares)
4. $\mathcal{L}_{no\lambda} = \{L \subseteq \Sigma^* \;:\; \lambda \notin L\}$
   (clase de los lenguajes que no contienen λ)