Parte III: Lenguajes y Autómatas

23 de noviembre de 2018

Repaso

Alfabeto Conjunto finito no vacío, habitualmente usamos Σ para referirnos a alfabetos.

Cadena (fijado el alfabeto) Secuencia finita de símbolos de Σ.

Lenguaje un subconjunto de Σ^*.

Lenguaje Regular puede ser reconocido por un AFD, o un AFN o un AFN-ϵ.

Hablemos de lenguajes no-regulares

Gracias al pumping lemma demostramos que

$$L_{()} = \{\alpha \mid \alpha \text{ tiene paréntesis balanceados}\}$$

no es un lenguaje regular.

Fenómenos similares en computación:

- lenguajes de programación: `{}`, `if`…`then`…`else`…
- lenguajes de intercambio de datos: json ({} y []), XML (etiquetas)

Las *gramáticas libres de contexto* son el formalismo para describir lenguajes con esas características.

Gramáticas libres de contexto

Una gramática para $0^n 1^n$:

$$A \rightarrow \mathtt{0}A\mathtt{1} \mid \epsilon$$

Una gramática para paréntesis balanceados:

$$S \rightarrow S\mathtt{(}S\mathtt{)}S \mid \epsilon$$

Una gramática para expresiones aritméticas con variables:

$$\begin{aligned}
E &\rightarrow N \mid I \mid E\mathtt{+}E \mid E\mathtt{*}E \mid \mathtt{(}E\mathtt{)} \\
I &\rightarrow \mathtt{a} \mid I\mathtt{0} \mid \ldots \mid I\mathtt{9} \\
N &\rightarrow \mathtt{1}D \mid \ldots \mid \mathtt{9}D \\
D &\rightarrow \mathtt{0}D \mid \ldots \mid \mathtt{9}D \mid \epsilon
\end{aligned}$$

Formalizando las Gramáticas Libres de Contexto

Una GLC está dado por los siguientes componentes $G = (V, T, P, S)$:

Alfabeto terminal Conjunto de símbolos sobre los que definimos el lenguaje.

$T_0 = \{0, 1\}, T_1 = \{(,)\}, T_2 = \{0, \ldots, 9, +, *, (,)\}$.

Variables (o símbolos no-terminales) Son las categorías sintácticas; cada una representará un lenguaje.

$V_0 = \{A\}, V_1 = \{S\}, V_2 = \{E, N, D, I\}$

Símbolo inicial Es una variable que determina el lenguaje que estamos generando.

$S_0 = A, S_1 = S, S_2 = E$

Reglas (o producciones) representan la definición recursiva del lenguaje: son de la forma $U \to \alpha$ donde $\alpha \in (V \cup T)^*$.

Decimos que U es la *cabeza* y α el *cuerpo*.

Comenzamos con la variable inicial y elegimos una de las reglas para esa variable y así sucesivamente:

$$
\begin{aligned}
&S \Rightarrow (S) \Rightarrow ((S)) \Rightarrow ((\epsilon)) \\
&E \Rightarrow E * E \Rightarrow I * E \Rightarrow I * (E) \Rightarrow a * (E) \\
&\quad \Rightarrow a * (N) \Rightarrow a * (ND) \Rightarrow a * (4D) \Rightarrow a * (42)
\end{aligned}
$$

Formalmente, si $\alpha, \beta \in (V \cup T)^*$ decimos que α *deriva* β si obtenemos β reemplazando en α una variable de V por el cuerpo de una producción $V \to \gamma$:

$$\alpha' V \alpha'' \Rightarrow \alpha' \gamma \alpha''$$

El lenguaje de una gramática

La clausura reflexiva-transitiva de $\Rightarrow$ nos permite definir el lenguaje de una gramática:

reflexiva $\alpha \Rightarrow^* \alpha$

transitiva Si $\alpha \Rightarrow^* \beta$ y $\beta \Rightarrow \gamma$, entonces $\alpha \Rightarrow^* \gamma$.

Volviendo al ejemplo podemos escribir $S \Rightarrow^* a * (42)$.

Sea $G = (V, T, P, S)$, su lenguaje está definido como

$$L(G) = \{\alpha \in T^* \mid S \Rightarrow^* \alpha\}$$

Observaciones sobre derivaciones

- Podemos elegir más de una variable para realizar una derivación:

$$a * E \Leftarrow I * E \Rightarrow I * (E)$$

  Se puede adoptar una estrategia para elegir siempre la primer variable de izquierda a derecha.

- Existe otra forma de representar las derivaciones, los *árboles de derivación*, que muestran que las gramáticas pueden ser ambiguas.

Árboles de derivación

nodos interiores variables de la gramática.

raiz el símbolo inicial.

hojas no-terminales o terminales o ϵ.

Un árbol donde todas las hojas sean terminales es una palabra de la gramática.

Si una palabra tiene dos árboles distintos, entonces la gramática es ambigua.

Si definen una gramática, asegúrense que sea no ambigua.