Escuela Técnica Superior de Ingeniería **Informática**

etsinf

DEPARTAMENTO DE SISTEMAS INFORMÁTICOS Y COMPUTACIÓN

Computabilidad y Complejidad

Práctica 1: Gramáticas Formales
(I – Definiciones básicas y representación)

Gramáticas Formales
(I- Definiciones básicas y representación)

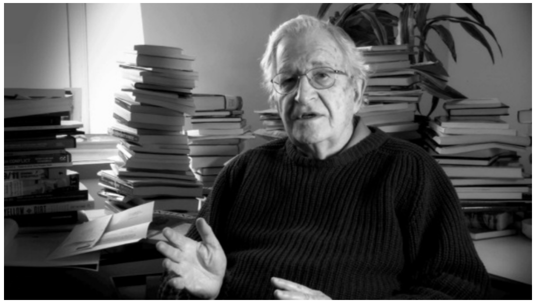
Índice:

Bibliografía Básica

- Introduction to Automata Theory, Languages and Computation (J.E. Hopcroft, R. Motwani, J.D. Ullman - Addison Wesley, 2001)
- Teoría de la computación (J. Glenn Brookshear - Addison Wesley Iberoamericana, 1993)

Gramáticas Formales

Una (breve) introducción



Noam Chomsky

Según la RAE, la gramática es "*la ciencia que estudia los elementos de una lengua y sus combinaciones.*" y "*el tratado de esta ciencia*". En nuestro caso, nos interesan las gramáticas desde el punto de vista de los lenguajes formales. Es decir que nos interesan las gramáticas conocidas como gramáticas generativas que según la RAE son aquellas " *… que tratan de formular una serie de reglas capaces de generar o producir todas las oraciones posibles y aceptables de un idioma*" (en nuestro caso, de un lenguaje formal. Nos referiremos a las gramáticas generativas para los lenguajes formales como gramáticas formales.

Los modelos de gramáticas generativas, en sus ideas básicas, siguen fundamentalmente la "teoría estándar" propuesta por Noam Chomsky hacia 1965 [1] (basándose en su propia tesis doctoral y trabajos anteriores como en [2]).

En el caso de las gramáticas formales, la formulación de este modelo permite establecer de forma rigurosa qué sentencias están correctamente formuladas en un lenguaje y cuáles no. Más aún, la decisión acerca de la corrección o no de una sentencia se puede realizar de forma automática y eficiente en algunos casos, lo que ha dado lugar al diseño efectivo de compiladores e intérpretes en el marco de la Informática. Por ejemplo, en la actualidad operamos con compiladores e intérpretes para lenguajes de programación, lenguajes de especificación, lenguajes de control de procesos, etc.

Bibliografía adicional

[1] N. Chomsky. *Aspects of the theory of syntax*. Cambridge, Massachusetts: MIT Press. 1965
[2] N. Chomsky. *Syntactic Structures*. The Hague: Mouton.1957

Gramáticas Formales

Definición formal

Una gramática G se define mediante la siguiente tupla:

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

donde

N y T son alfabetos de símbolos auxiliares y terminales respectivamente con $N \cap T = \emptyset$

S ∈ N es el símbolo inicial o axioma

P es un conjunto finito de producciones. Una producción la definiremos como un par (α,β) donde

α es la parte izquierda ó antecedente de la producción y es una cadena sobre $(N \cup T)^* N (N \cup T)^*$

β es la parte derecha ó consecuente de la producción y es una cadena sobre $(N \cup T)^*$

Notación

- la producción (α,β) la escribiremos como $\alpha \to \beta$ (leído " … *alfa produce beta*")
- si nos encontramos con un conjunto de producciones que comparten antecedente (α,β_1), (α,β_2),…, (α,β_n) entonces escribiremos $\alpha \to \beta_1 \mid \beta_2 \mid \ldots \mid \beta_n$

Notación

Habitualmente escribiremos las gramáticas de forma compacta, especificando únicamente sus producciones y siguiendo el convenio establecido a continuación

(a) los símbolos del abecedario en mayúscula se referirán a símbolos auxiliares

(b) los símbolos del abecedario en minúscula, los dígitos y cualquier otro símbolo especial se referirán a símbolos terminales

(c) el axioma se escribirá como S

cualquier alteración del anterior convenio se establecerá de forma explícita.

Gramáticas Formales

Ejemplo 1

$$S \to aBSc \mid abc$$

$$Ba \to aB$$

$$Bb \to bb$$

Ejemplo 2

$$S \to U \mid V$$

$$U \to TaU \mid TaT$$

$$V \to TbV \mid TbT$$

$$T \to aTbT \mid bTaT \mid \lambda$$

Ejemplo 3

$$S \to A$$

$$A \to aA \mid B$$

$$B \to bB \mid C$$

$$C \to cC \mid \lambda$$

Gramáticas Formales

Lenguaje generado por una gramática

Relación de derivación directa

Sea G=(N,T,P,S) una gramática y α,β dos cadenas sobre $(N \cup T)^*$ Diremos que α deriva directamente en β de acuerdo con G, denotándolo por $\alpha \underset{G}{\Rightarrow} \beta$ si se cumple que $\alpha=\alpha_1\alpha_2\alpha_3$, $\beta=\alpha_1\beta_1\alpha_3$ y $\alpha_2 \to \beta_1 \in P$

Relación de derivación

Sea G=(N,T,P,S) una gramática y α,β dos cadenas sobre $(N \cup T)^*$ Diremos que α deriva en β de acuerdo con G, denotándolo por $\alpha \overset{*}{\underset{G}{\Rightarrow}} \beta$ si se cumple una de las dos siguientes condiciones:

(a) $\alpha=\beta$

(b) existe una cadena γ de forma que $\alpha \underset{G}{\Rightarrow} \gamma \overset{*}{\underset{G}{\Rightarrow}} \beta$

Formas sentenciales, palabras y lenguajes

Sea G=(N,T,P,S) una gramática. Diremos que α es …

(a) una forma sentencial de G si $\alpha \in (N \cup T)^*$ y $S \overset{*}{\underset{G}{\Rightarrow}} \alpha$

(b) una palabra de G si $\alpha \in T^*$ y $S \overset{*}{\underset{G}{\Rightarrow}} \alpha$

Lenguaje generado por una gramática G. Equivalencia entre gramáticas

Sea G=(N,T,P,S) una gramática. El lenguaje generado por G lo definimos como el siguiente conjunto

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

Dos gramáticas G_1 y G_2 diremos que son equivalentes si $L(G_1)=L(G_2)$.

Notación: Los símbolos $\overset{*}{\underset{G}{\Rightarrow}}$ y $\underset{G}{\Rightarrow}$ los reescribiremos como $\overset{*}{\Rightarrow}$ y $\Rightarrow$ siempre que la gramática G quede sobreentendida

Gramáticas Formales

Ejemplo 1

$$S \to aBSc \mid abc$$
$$Ba \to aB$$
$$Bb \to bb$$

Genera el lenguaje

$$\{a^n b^n c^n : n \geq 1\}$$

Ejemplo 2

$$S \to U \mid V$$
$$U \to TaU \mid TaT$$
$$V \to TbV \mid TbT$$
$$T \to aTbT \mid bTaT \mid \lambda$$

Genera el lenguaje de cadenas formadas por símbolos *a* y *b* donde el número de símbolos *a* es distinto del número de símbolos *b*

Ejemplo 3

$$S \to A$$
$$A \to aA \mid B$$
$$B \to bB \mid C$$
$$C \to cC \mid \lambda$$

Genera el lenguaje de cadenas pertenecientes a la expresión regular *a*b*c**

Gramáticas Formales

Representación en Mathematica

Recordemos que las cadenas en Mathematica las representamos mediante listas. Por ejemplo, la cadena abbba se representa como {a,b,b,b,a} y la cadena vacía λ se representa como la lista vacía {}

Sea la gramática G=(N,T,P,S). La representación de G en *Mathematica*, la haremos mediante una lista de cuatro elementos: La lista N, la lista T, la lista P y el elemento S.

G={N,T,P,S}

A su vez, P será una lista de listas, donde por cada antecedente común de las producciones, tendremos una lista

$$S \to aBSc \mid abc$$

$$Ba \to aB$$

$$Bb \to bb$$

P={producciones_de_S, producciones_de_Ba, producciones_de_Bb}

Gramáticas Formales

Representación en Mathematica

$$S \to aBSc \mid abc$$
$$Ba \to aB$$
$$Bb \to bb$$

P={producciones_de_S, producciones_de_Ba, producciones_de_Bb}

Las producciones de cada antecedente común se representarán mediante una lista de dos listas (una lista para la parte izquierda común y una lista con las partes derechas)

$$S \to aBSc \mid abc$$

producción

$$S \to aBSc$$

producción

$$S \to abc$$

producciones_de_S={{S},{{a,B,S,c},{a,b,c}}}

parte izquierda común
(una lista con el antecedente)

partes derechas de las producciones
(una lista con tantas listas como
partes derechas existan)

Gramáticas Formales

Ejemplo 1

$$S \to aBSc \mid abc$$
$$Ba \to aB$$
$$Bb \to bb$$

{{S,B},{a,b,c},{{{S},{{a,B,S,c},{a,b,c}}},{{B,a},{{a,B}}},{{B,b},{{b,b}}}},S}

Ejemplo 2

$$S \to U \mid V$$
$$U \to TaU \mid TaT$$
$$V \to TbV \mid TbT$$
$$T \to aTbT \mid bTaT \mid \lambda$$

{{S,U,V,T},{a,b},{{{S},{{U},{V}}},{{U},{{T,a,U},{T,a,T}}},{{V},{{T,b,V},{T,b,T}}},{{T},{a,T,b,T},{b,T,a,T},{{}}}}},S}

Ejemplo 3

$$S \to A$$
$$A \to aA \mid B$$
$$B \to bB \mid C$$
$$C \to cC \mid \lambda$$

{{S,A,B,C},{a,b,c},{{{S},{{A}}},{{A},{{a,A},{B}}},{{B},{{b,B},{C}}},{{C},{{c,C},{{}}}}},S}

Gramáticas Formales

La Jerarquía de Chomsky

La jerarquía de Chomsky es un marco de estudio de los lenguajes formales basado en una taxonomía de las gramáticas en función de la forma que toman las producciones. Se proponen cuatro grandes familias de gramáticas que, a su vez, definen cuatro familias de lenguajes.

Gramáticas de tipo 0 ó no restringidas

No tienen ningún tipo de restricción

Gramaticas de tipo 1 ó sensibles al contexto

Las producciones toman la forma

$$\alpha A\beta \rightarrow \alpha\gamma\beta \quad \alpha,\beta \in (N \cup T)^*, A \in N, \gamma \in (N \cup T)^+$$

Adicionalmente se permite la regla $S \rightarrow \lambda$ siempre que S no aparezca como consecuente en ninguna producción

Gramáticas de tipo 2 ó de contexto libre ó incontextuales

Las producciones toman la forma

$$A \rightarrow \alpha \quad \alpha \in (N \cup T)^*, A \in N$$

Gramáticas de tipo 3 ó regulares que se subdividen en dos tipos

Gramáticas lineales por la derecha

Las producciones toman la forma

$$A \rightarrow aB \quad a \in T, A, B \in N$$

$$A \rightarrow a \quad a \in T, A \in N$$

$$A \rightarrow B \quad A, B \in N$$

$$A \rightarrow \lambda \quad A \in N$$

Gramáticas lineales por la izquierda

Las producciones toman la forma

$$A \rightarrow Ba \quad a \in T, A, B \in N$$

$$A \rightarrow a \quad a \in T, A \in N$$

$$A \rightarrow B \quad A, B \in N$$

$$A \rightarrow \lambda \quad A \in N$$

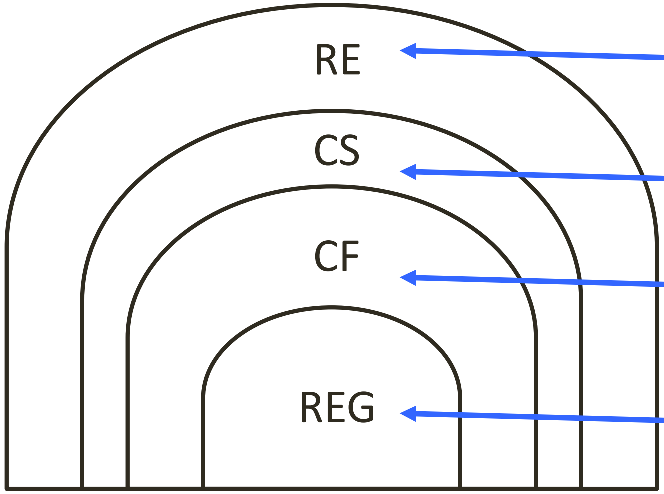
Gramáticas Formales

La Jerarquía de Chomsky

Desde el punto de vista de las clases de lenguajes, los cuatro tipos de gramáticas definidas en la jerarquía de Chomsky definen cuatro familias de lenguajes, atendiendo al siguiente criterio:

“Un lenguaje es de tipo i si existe una gramática de tipo i que lo genera”

De esta forma y desde el punto de vista de las clases de lenguajes que originan los cuatro tipos de gramáticas, la jerarquía de Chomsky establece una relación de clases de lenguajes que obedece al siguiente esquema



lenguajes recursivamente enumerables

lenguajes sensibles al contexto

lenguajes incontextuales

lenguajes regulares

Gramáticas Formales

Ejemplo 1

$$S \to aBSc \mid abc$$
$$Ba \to aB$$
$$Bb \to bb$$

Gramática de tipo 0 (existe una gramática equivalente de tipo 1)

Ejemplo 2

$$S \to U \mid V$$
$$U \to TaU \mid TaT$$
$$V \to TbV \mid TbT$$
$$T \to aTbT \mid bTaT \mid \lambda$$

Gramática de tipo 2

Ejemplo 3

$$S \to A$$
$$A \to aA \mid B$$
$$B \to bB \mid C$$
$$C \to cC \mid \lambda$$

Gramática de tipo 3 lineal por la derecha

Gramáticas Formales

Actividades propuestas

1. Dada una gramática incontextual, diremos que un símbolo auxiliar es *directamente generativo* si aparece como antecedente de una producción cuyo consecuente es una cadena de símbolos terminales (incluida la cadena vacía). Implemente un módulo *Mathematica* que, tomando como entrada una gramática incontextual, obtenga como salida una lista con aquellos símbolos auxiliares de la gramática *directamente generativos*.

2. Dada una gramática incontextual, diremos que un símbolo auxiliar es *directamente no generativo* si en las producciones donde aparece como antecedente aparece también en el consecuente. Implemente un módulo *Mathematica* que, tomando como entrada una gramática incontextual, obtenga como salida una lista con aquellos símbolos auxiliares de la gramática *directamente no generativos*.

3. Dada una gramática incontextual, diremos que está en Forma Normal de Greibach si sus producciones toman la siguiente forma

$$A \to \alpha\beta : A \in N, \quad \alpha \in T, \quad \beta \in N^*$$

   Implemente un módulo en *Mathematica* que, tomando como entrada una gramática incontextual devuelva True si está en Forma Normal de Greibach y False en caso contrario.

4. Implemente un módulo en *Mathematica* que, tomando como entrada una gramática cuyas producciones están en una de las dos formas que se indican al final, obtenga como salida una gramática equivalente lineal por la derecha

$$A \to \alpha B \quad \alpha \in \mathrm{T}^*, A, B \in N$$
$$A \to \alpha \quad \alpha \in \mathrm{T}^*, \mathrm{A} \in N$$