Lenguajes Formales, Auómatas y Computabilidad

Clase Teórica

Lenguajes libres de contexto,
propiedades de clausura, algoritmos de decisión

Primer Cuatrimestre 2025

Bibliografía: Capítulos 6 y 7, *Introduction to Automata Theory, Languages and Computation*, J. Hopcroft, R. Motwani, J. Ullman, Second Edition, Addison Wesley, 2001.

Algoritmos de decisión

Teorema

Hay un algoritmo para decidir si un lenguaje libre de contexto es vacío.

Sea n la constante del Lema de Pumping. L es vacío si y solo si ninguna palabra de longitud menor que n pertenece a L.

Si hubiera una de longitud n o más, por el Lema de Pumping también habría una de longitud menor que n.

¿Qué hace este algoritmo?

Input $G = (N, T, P, S)$ libre de contexto
$N_0 = \emptyset$
repetir
  $i = i + 1$
  $N_i = \{A : A \to \alpha \in P, \alpha \in (N_{i-1} \cup T)^*\} \cup N_{i-1}$
hasta que $N_i = N_{i-1}$
Si $S \in N_i$, output **SÍ**.

Algoritmos de decisión

Teorema

Hay un algoritmo para decidir si un lenguaje libres de contexto es finito.

Demostración

Sea n la constante del Lema de Pumping. L es finito si y solo si ninguna palabra de longitud entre n y $2n-1$.

(izquierda a derecha) Supongamos L es finito pero tiene una palabra de longitud entre n y $2n-1$. Lema de Pumping también L tiene infinitas de longitud mayor que n. Contradicción.

(derecha a izquierda) Supongamos L no tiene ninguna palabra de longitud entre n y $2n-1$ pero es infinito. Sea w la palabra en L más corta de longitud mayor o igual que $2n$. El lema de Pumping afirma que existe una palabra más corta que w que está en L:
Hay una factorización $w = rxyzs$ con $|xyz| \leq n$, $|xz| \geq 1$ y tenemos que $rx^0yz^0s = rys$ está en L. Dado que $|xz| \geq 1$ y $|xyz| \leq n$

$$|w| - n \leq |rys| = |w| - |xz| \leq |w| - 1$$

Si $|w| = 2n$, $|rys|$ es entre n y $2n-1$ contradiciendo la suposición.
Si $|w| > 2n$, $|rys|$ es estrictamente menor que w. O bien $|rys|$ es entre n y $2n-1$ y llegamos a contradicción, o bien $|rys|$ es mayor o igual que $2n$ contradiciendo que w era la más corta de longitud mayor o igual que $2n$.
$\square$

Algoritmos de decisión

Teorema

Hay un algoritmo para decidir la pertenecia de una palabra a un lenguaje libre de contexto.

Demostración: Algoritmo de parsing CYK o Earley (no lo veremos).

Resumen: Decisión de lenguajes libres de contexto

Sea G gramática libre de contexto y sea $L = L(G)$.

Hay algoritmos para :
$L = \emptyset$?
L finito?
L infinito?
$w \in L$? en tiempo cúbico en la longitud de w (algoritmo CYK).

No hay algoritmos para:
L es regular?
G libre de contexto es ambigua?
$L_1 = L_2$?
$L = \Sigma^*$?
$L_1 \subseteq L_2$?
$L_1 \cap L_2 = \emptyset$?

Propiedades de Clausura

Teorema

Los lenguajes libres de contexto están cerrados por unión, concatenación, reversa y clausura de Kleene.

No están cerrados por intersección, diferencia, complemento.

Sin embargo la intersección de libre de contexto con regular es libre de contexto.

Demostración del teorema

union (gramática)

Supongamos $G_1 = (N_1, T_1, P_1, S_1)$ *y* $G_2 = (N_2, T_2, P_2, S_2)$.
Definimos $G = (N_1 \cup N_2 \cup \{S\}, T_1 \cup T_2, P, S)$ *donde*
$P = P_1 \cup P_2 \cup \{S \to S_1 | S_2\}$

concatenacion (gramática)

Supongamos $G_1 = (N_1, T_1, P_1, S_1)$ *y* $G_2 = (N_2, T_2, P_2, S_2)$.
Definimos $G = (N_1 \cup N_2 \cup \{S\}, T_1 \cup T_2, P, S)$ *donde*
$P = P_1 \cup P_2 \cup \{S \to S_1 S_2\}$

clausura Kleene (gramática)

Supongamos $G = (N, T, P, S)$
Definimos $G' = (N \cup \{S'\}, T, P', S')$ *donde*
$P' = P \cup \{S' \to SS' | \lambda\}$

intersección con lenguaje regular (autómata de pila para lenguaje intersección)

Demostración del teorema

reversa (gramática invirtiendo cuerpo gramática)

Supongamos $G = (V, T, P, S)$ *es libre de contexto,* $L = L(G)$*.*
Construimos $G' = (V, T, P', S)$ *para* L^R *así:*

Para cada producción $X \to \alpha$ *en* P *ponemos* $X \to \alpha^R$ *en* P'*.*

Supongamos P *tiene* $S \to uXv$ *y* $X \to \alpha$*.*
Entonces P' *tiene* $S \to v^R X u^R$ *y* $X \to \alpha^R$*.*
Luego $S \underset{G}{\Rightarrow} u\alpha v$ *y* $S \underset{G'}{\Rightarrow} v^R \alpha^R u^R$*.*
Dado que $v^R \alpha^R u^R = (u\alpha v)^R$*, tenemos* $S \underset{G'}{\Rightarrow} (u\alpha v)^R$*.*

Demostración del teorema

No están clausurados por :

intersección

Sean los lenguajes libres de contexto
$L_1 = \{a^i b^j c^j\}$ *y* $L_2 = \{a^i b^i c^j\}$ *.*
Notemos que $L_1 \cap L_2 = \{a^i b^i c^i\}$ *no es libre de contexto.*

complemento

Supongamos que el complemento fuera libre de contexto.
Entonces
$L_1 \cap L_2 = \overline{\overline{L_1} \cup \overline{L_2}}$ *sería libre de contexto.*

diferencia

Si lo fuera entonces $\overline{L} = \Sigma^* - L$ *deberia ser libre de contexto*

□

Cota en la cantidad de pasos en una derivación

Una gramática no es recursiva a izquierda si no tiene derivaciones $A \underset{L}{\overset{+}{\Rightarrow}} A\alpha$, para ningún $\alpha \in (V_N \cup V_T)^*$.

Lema (Lema 4.1 Aho-Ullman vol. 1)

Sea $G = (N, T, P, S)$ libre de contexto y no recursiva a izquierda. Existe una constante c tal que si $A \underset{L}{\overset{i}{\Rightarrow}} wB\alpha$ y entonces $i \leq c^{|w|+2}$.

Se puede demostrar un resultado mucho más ajustado, con i polinomial en $|w|$, pero la misma constante.

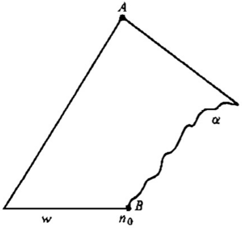
En particular, el resultado anterior vale para w igual a la cadena vacía.

Corolario

Sea G libre de contexto y no recursiva. Existe una constante c tal que para todo par de no terminales A, B, si $A \underset{L}{\overset{i}{\Rightarrow}} B\alpha$ entonces $i \leq c^2$.

Demostración del lema

Llamemos $k = |V_N|$. Consideremos el ábol de la derivación más a la izquierda para $A \underset{L}{\overset{i}{\Rightarrow}} wB\alpha$.



Sea n_0 el nodo con etiqueta B en la derivación $A \underset{L}{\overset{i}{\Rightarrow}} wB\alpha$. Por ser la derivación más a la izquierda, todos los caminos a la derecha del camino desde la raíz a n_0 son más cortos, o del mismo largo. Sea $A = X_0 \frown X_1 \frown X_{(n+2)k} = B$. el camino desde la raíz a n_0, Separemoslos en $(n+2)$ segmentos de $k+1$ nodos,

$$A = X_0 \frown \ldots \frown X_k, \quad X_k \frown \ldots \frown X_{2k}, \ldots, \quad X_{(n+1)k} \frown \ldots \frown X_{(n+2)k} = B.$$

notar que se repite el final de uno con el principio del siguiente.

Cosideremos los $(n+2)$ subarboles de derivación (con β apropiados)
$A = X_0 \underset{L}{\overset{k}{\Rightarrow}} \beta_1 X_k, \quad X_k \underset{L}{\overset{k}{\Rightarrow}} \beta_2 X_{2k}, \quad \ldots, X_{(n+1)k} \underset{L}{\overset{k}{\Rightarrow}} \beta_{(n+2)k} X_{(n+2)k}$.
Es imposible que cada uno produzca uno o más símbolos de wB, porque $|wB| = n+1$. Al menos uno no produce ningun símbolo de wB,

$$X_{jk} \underset{L}{\Rightarrow} \beta_{jk+1} X_{jk+1} \underset{L}{\Rightarrow} \ldots \underset{L}{\Rightarrow} \beta_{(j+1)k} X_{(j+1)k}$$

y deriva solamente λ. Cada uno de $X_{jk}, \beta_{jk+1} X_{jk+1}, .., \beta_{(j+1)k} X_{(j+1)k}$ empiezan con un símbolo de V_N, son en total $k+1$, Necesariamente hay dos que empiezan con el mismo símbolo. Pero esto contradice que la gramática no es recursiva a izquierda. Entonces nuestra suposición de que el camino de la raíz a n_0 tiene $(n+2)$ segmentos de $(k+1)$ nodos es imposible. Concluimos que su longitud es menor que $(n+2)k$.
Sea ℓ el máximo número de símbolos en la parte derecha de una producción de la gramática. La cantidad de nodos del árbol de derivación es a lo sumo $\ell^{k(n+2)}$. Por lo tanto, $A \underset{L}{\overset{i}{\Rightarrow}} wB\alpha$ con $i \leq \ell^{k(n+2)}$.
Para finalizar la demostración basta tomar $c = \ell^k$. $\square$

Autómata de Pila Enriquecidos

Autómatas con un contador.
Son autómatas donde el alfabeto de pila tiene dos símbolos: Z_0 y I.
En cada transición pueden revisar si el contador es Z_0, o no, y pueden incrementar o decrementarlo.

Los lenguajes reconocidos por autómatas contadores incluyen a todos los lenguajes regulares pero son un subconjunto propio de los lenguajes reconocidos por autótas de pila.

Autómata de Pila Enriquecidos

Los autómatas de pila de dos vías, tienen una pila, control finito y una cinta cuya cabeza puede moverse en ambas direcciones (Aho, Hopcroft, Ullman 1968, and Gray, Harrison, Ibarra 1967). Estos autómatas tiene menos poder computacional que una máquina de Turing.

Los autómatas con dos o más pilas son equivalentes a una máquina de Turing (Aho Ullman Teoremas 8.13, 8.14 y 8.15).

Los autómatas con tres contadores son equivalentes a una máquina de Turing (Aho Ullman Teoremas 8.13, 8.14 y 8.15).

Los autómatas con dos contadores son equivalentes a una máquina de Turing (Aho Ullman Teoremas 8.13, 8.14 y 8.15).

Los autómatas con una cola son equivalentes a autómatas con dos pilas, por lo tanto, equivalentes a una máquina de Turing (Bisbay 1959).

Preguntas

1. Sea L un lenguaje regular. Demostrar que todas las palabras de L validan el Lema de Pumping para lenguajes libres de contexto.
2. Mostrar que $L = \{a^p : p \text{ es número primo}\}$ no es libre de contexto. Ayuda: . Asumir L es libre de contexto, y n es la longitud dada por el Lema de Pumping. Sea m el primer primo mayor o igual que n, considerar $\alpha = a^m$ y bombear $m + 1$ veces.
3. Demostrar que un lenguaje regular intersección un lenguaje libre de contexto es libre de contexto. Ayuda: Definir el autómata de pila que lo reconoce.
4. Demostrar que hay lenguajes que pueden ser reconocidos con un autómata con dos pilas pero no pueden ser reconocidos con un autómata con una sola pila. Ayuda: los ejemplos en esta clase.