Chapter 7

上下文无关语言的性质

7.1 上下文无关语言的泵引理

定理 1 (上下文无关语言的泵引理). 设 L 是任意 CFL, 那么存在常数 N, 它仅依赖于 L, 使得若 $z \in L$, $|z| \geq N$, 则总可以将 z 写成 $z = uvwxy$, 满足:

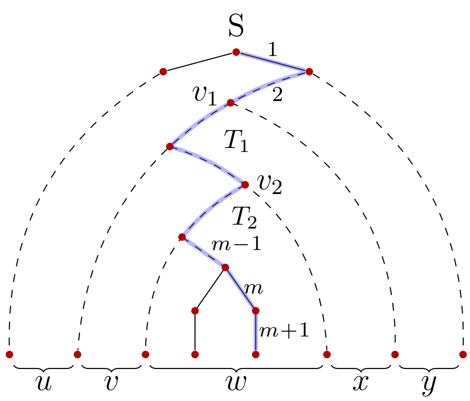
(1) $|vwx| \leq N$;

(2) $vx \neq \varepsilon$(或 $|vx| \neq 0$);

(3) 对任意 $i \geq 0$, $uv^iwx^iy \in L$.

证明. 设 $G = (V, T, P, S)$ 是接受 $L - \{\varepsilon\}$ 的 CNF 文法. 在 CNF 文法的派生树中, 若最长路径为 k, 则产物的长度最多为 2^{k-1}. 设 G 变元数为 m, $N = 2^m$, 那么若有 $z \in \mathbf{L}(G)$, $|z| \geq N$, 则 z 的派生树中最长路径长度至少也是 $m+1$, 这个路径上有至少 $m+2$ 个节点, 除最后一个节点外, 其余标记都是变元. 只考虑在接近树底部连续的 $m+1$ 个变元标记, 其中至少有两个是相同的.

如果这两个节点分别是 v_1 和 v_2, 标记均为 A, v_1 比 v_2 更接近树根. 设以 v_1 为根的子树为 T_1, 它的产物 z_1 长度不会超过 2^m, 因为 T_1 最长路径不超过 $m+1$. 设以 v_2 为根的子树为 T_2 产物为 w, 那么 $z_1 = vwx$. 而且 v 和 x 不能同时为空, 因为 z_1 派生的第一个产生式必须是 $A \to BC$, T_2 不是完全处于 B 中就是完全处于 C 中, 而 B 或 C 都至少产生一个终结符.



那么可以得到

$$A \stackrel{*}{\Rightarrow} vAx \quad \text{和} \quad A \stackrel{*}{\Rightarrow} w$$

而且 $|vwx| = |z_1| \leq 2^m = N$. 所以对任意 $i \geq 0$, $A \stackrel{*}{\Rightarrow} v^iwx^i$. 那么串 z 可以写成 $uvwxy$, u 和 y 为某个串, 即 $S \stackrel{*}{\Rightarrow} uAy \stackrel{*}{\Rightarrow} uv^iwx^iy$. $\square$

7.1.1 CFL 泵引理的应用

示例

证明 $L = \{0^n1^n2^n \mid n \geq 1\}$ 不是上下文无关语言.

证明. 假设 L 是上下文无关的, 那么存在整数 N. 取 $z = 0^N1^N2^N$. 由泵引理, $z = uvwxy$, 其中 $|vwx| \leq N$, $vx \neq \varepsilon$. 如果 vwx 只包含 0, 1 或 2, 那么 uwy 不在 L 中; vwx 若只包含 0 和 1, 或只包含 1 和 2, uwy 也不在 L 中. 而由于泵引理 $uwy = uv^0wx^0y \in L$, 因此假设不成立, L 不是上下文无关的. □

证明 $L = \{a^ib^jc^id^j \mid i \geq 1 \text{ and } j \geq 1\}$ 不是上下文无关的. (取 $z = a^nb^nc^nd^n$.)

证明 $L = \{ww \mid w \in \{0,1\}^*\}$ 不是上下文无关的.

(错误的) *证明.* 假设 L 是 CFL. 取 $z = 0^N10^N1$, 那么 $z = uvwxy$ 为

$$z = \overbrace{\underbrace{00\cdots00}_{u}\ \underbrace{0}_{v}\ \underbrace{1}_{w}}^{0^N1}\overbrace{\underbrace{0}_{x}\ \underbrace{00\cdots01}_{y}}^{0^N1}$$

则对任意 $i \geq 0$, 有 $uv^iwx^iy \in L$, 满足泵引理. □

(正确的) *证明.* 假设 L 是 CFL. 取 $z = 0^N1^N0^N1^N$, 那么 $z = uvwxy$ 时

(1) 若 vwx 在 z 中点的任意一侧, uv^0wx^0y 显然不可能属于 L;

(2) 若 vwx 中包括 z 中点, 那么 uv^0wx^0y 只能形如 $0^N1^i0^j1^N$, 也不可能属于 L.

所以假设不成立. □

7.2 上下文无关语言的封闭性

7.2.1 代换

代换 (*substitution*) 是映射 $s : \Sigma \mapsto 2^{\Gamma^*}$. Σ 中的一个字符 a 在 s 的作用下成为语言 L_a, 即

$$s(a) = L_a.$$

再推广 s 到 Σ 的字符串:

$$\begin{aligned} &(1) \quad s(\varepsilon) = \varepsilon \\ &(2) \quad s(xa) = s(x)s(a) \end{aligned}$$

再推广 s 到 Σ 的语言 L:

$$s(L) = \bigcup_{x \in L} s(x).$$

定理 2. *上下文无关语言在代换下封闭.*

说明: 如果 L 是 Σ 上的上下文无关语言, s 是 Σ 上的代换, 且每个 $a \in \Sigma$, $s(a)$ 都是 CFL, 那么 $s(L)$ 是 CFL.

证明. **文法构造**: 若任意 $a \in \Sigma$, $s(a)$ 都是 CFL, 那么设 $s(a)$ 的文法为 $G_a = (V_a, T_a, P_a, S_a)$; 设 L 的文法 $G = (V, T, P, S)$. 那么 $s(L)$ 的文法可以构造为 $G' = (V', T', P', S)$:

(1) $V' = (\bigcup_{a \in T} V_a) \cup V$

(2) $T' = \bigcup_{a \in T} T_a$

(3) P' 包括:

  (i) 每个 P_a 中的产生式;

  (ii) P 的产生式, 但要替换产生式中的终结符 a 为 S_a.

那么, 需证明 $s(L) = \mathbf{L}(G')$.

充分性 ($s(L) \subseteq \mathbf{L}(G')$): 对 $\forall w \in s(L)$, 那么一定存在某个 $x \in L$ 使 $w \in s(x)$. 设 $x = a_1 a_2 \cdots a_n$ 即

$$w \in s(x) = s(a_1)s(a_2)\cdots s(a_n),$$

那么 w 可以分为 $w = w_1 w_2 \cdots w_n$ 且 $w_i \in s(a_i)$, 即 $S_{a_i} \overset{*}{\underset{G_{a_i}}{\Rightarrow}} w_i$. 由于 $S \overset{*}{\underset{G}{\Rightarrow}} x = a_1 a_2 \cdots a_n$, 所以

$$S \overset{*}{\underset{G'}{\Rightarrow}} S_{a_1} S_{a_2} \cdots S_{a_n} \overset{*}{\underset{G'}{\Rightarrow}} w_1 w_2 \cdots w_n = w,$$

所以 $w \in \mathbf{L}(G')$.

必要性 ($\mathbf{L}(G') \subseteq s(L)$): 对 $\forall w \in \mathbf{L}(G')$, 有 $S \overset{*}{\underset{G'}{\Rightarrow}} w$, 又因为 w 中每个终结符仅能由某个 S_a 派生出来, 所以存在仅由 S_a 构成的句型 α, 有

$$S \overset{*}{\underset{G'}{\Rightarrow}} \alpha \overset{*}{\underset{G'}{\Rightarrow}} w.$$

不妨设 $\alpha = S_{a_1} S_{a_2} \cdots S_{a_n}$, 那么因为 $S \overset{*}{\underset{G'}{\Rightarrow}} \alpha$, 所以

$$S \overset{*}{\underset{G}{\Rightarrow}} a_1 a_2 \cdots a_n$$

那么 $x = a_1 a_2 \cdots a_n \in L$. 而又因为 $\alpha = S_{a_1} S_{a_2} \cdots S_{a_n} \overset{*}{\underset{G'}{\Rightarrow}} w$, 所以 w 可以分为 $w = w_1 w_2 \cdots w_n$, 且对 $i = 1, 2, \cdots, n$ 有

$$S_{a_i} \overset{*}{\underset{G'}{\Rightarrow}} w_i,$$

所以 $w_i \in s(a_i)$, 那么

$$w = w_1 w_2 \cdots w_n \in s(a_1)s(a_2)\cdots s(a_n) = s(a_1 a_2 \cdots a_n) = s(x),$$

所以 $w \in s(L)$. □

示例

设 $L = \{w \mid w\text{有相等个数的}a\text{和}b\}$, 代换 $s(a) = L_a = \{0^n 1^n \mid n \geq 1\}$, $s(b) = L_b = \{ww^R \mid w \in (0+1)^*\}$, 求 $s(L)$ 的文法.

设计 L 的文法为: $S \to aSbS \mid bSaS \mid \varepsilon$

设计 L_a 的文法为: $S_a \to 0S_a1 \mid 01$

设计 L_b 的文法为: $S_b \to 0S_b0 \mid 1S_b1 \mid \varepsilon$

那么 $s(L)$ 的文法为:

$S \to S_a S S_b S \mid S_b S S_a S \mid \varepsilon$
$S_a \to 0S_a1 \mid 01$
$S_b \to 0S_b0 \mid 1S_b1 \mid \varepsilon$

7.2.2 并, 连接, 闭包, 同态/逆同态, 反转

定理 3. 上下文无关语言在并, 连接, 闭包, 正闭包, 同态运算下封闭.

证明. 若 $\Sigma = \{1, 2\}$, 语言 $\{1, 2\}$, $\{12\}$, $\{1\}^*$ 和 $\{1\}^+$ 显然都是 CFL. 设 L_1 和 L_2 是任意的 CFL, 并定义代换 $s(1) = L_1$, $s(2) = L_2$, 那么:

(1) 因为 $s(\{1, 2\}) = s(1) \cup s(2) = L_1 \cup L_2$, 所以并运算下封闭;

(2) 因为 $s(\{12\}) = s(12) = s(\varepsilon)s(1)s(2) = L_1L_2$, 所以连接运算下封闭;

(3) 因为

$$\begin{aligned} s(\{1\}^*) &= s(\{\varepsilon, 1, 11, 111, \cdots\}) \\ &= s(\varepsilon) \cup s(1) \cup s(11) \cup s(111) \cup \cdots \\ &= s(\varepsilon) \cup s(1) \cup s(1)s(1) \cup s(1)s(1)s(1) \cup \cdots \\ &= \{\varepsilon\} \cup L_1 \cup L_1L_1 \cup L_1L_1L_1 \cup \cdots \\ &= (s(1))^* = L_1^* \end{aligned}$$

所以闭包运算下封闭 (正比包, 同理).

若 h 是 Σ 上的同态, L 是 Σ 上的 CFL, 对 $\forall a \in \Sigma$ 令代换 $s(a) = \{h(a)\}$, 则

$$h(L) = \{h(w) \mid w \in L\} = \bigcup_{w \in L} \{h(w)\} = \bigcup_{w \in L} s(w) = s(L),$$

所以在同态运算下封闭. □

也可以使用文法来证明 CFL 并, 连接, 闭包的封闭性.

证明. 若 L_1 和 L_2 是 CFL, 那么设文法分别为 $G_1 = (V_1, T_1, P_1, S_1)$ 和 $G_2 = (V_2, T_2, P_2, S_2)$. 那么

(1) $L_1 \cup L_2$ 的文法为

$$G_{union} = (V_1 \cup V_2, T_1 \cup T_2, P_1 \cup P_2 \cup \{S \to S_1 \mid S_2\}, S);$$

(2) L_1L_2 的文法为

$$G_{concat} = (V_1 \cup V_2, T_1 \cup T_2, P_1 \cup P_2 \cup \{S \to S_1S_2\}, S);$$

(3) L_1^* 的文法为

$$G_{closure} = (V_1, T_1, P_1 \cup \{S \to S_1S_1 \mid \varepsilon\}, S).$$

再证明所构造文法的正确性, 略. □

定理 4 (反转)**.** *如果 L 是 CFL, 那么 L^R 也是 CFL.*

证明. 设 L 的文法 $G = (V, T, P, S)$, 构造文法 $G' = (V, T, \{A \to \alpha^R \mid A \to \alpha \in P\}, S)$, 则 $L(G') = L^R$. 证明略. □

定理 5. *CFL 在逆同态下封闭.*

证明. (构造部分) 已知 L 是字母表 Δ 上的 CFL, h 是 Σ 到 Δ^* 的同态. 设 PDA $P = (Q, \Delta, \Gamma, \delta, q_0, Z_0, F)$ 有 $L = \mathbf{L}(P)$. 构造识别 $h^{-1}(L)$ 的 PDA P' 使用缓冲暂存 a ($a \in \Sigma$) 的同态串 $h(a)$, 然后利用 P 的状态和缓冲中未消耗的 $h(a)$, 即其后缀, 形成的二元组作为 P' 的当前状态. 构造如下

$$P' = (Q', \Sigma, \Gamma, \delta', [q_0, \bar{\varepsilon}], Z_0, F \times \{\bar{\varepsilon}\})$$

其中

(1) 有限状态集 $Q' \subset Q \times \Delta^*$ 中的状态为 $[q, \overline{x}]$, 用 q 模拟 P 的状态, $\overline{x}$ 模拟缓冲;

(2) 设 $q \in Q$, 那么 δ' 定义如下:

(i) $\forall [q, \bar{\varepsilon}] \in Q \times \{\bar{\varepsilon}\}$, $\forall a \in \Sigma$, $\forall X \in \Gamma$

$$\delta'([q, \bar{\varepsilon}], a, X) = \{([q, h(a)], X)\}$$

(ii) 若 $\delta(q, \overline{a}, X) = \{(p_1, \beta_1), (p_2, \beta_2), \cdots, (p_k, \beta_k)\}$, 则

$$\delta'([q, \overline{ax}], \varepsilon, X) = \{([p_1, \overline{x}], \beta_1), ([p_2, \overline{x}], \beta_2), \cdots, ([p_k, \overline{x}], \beta_k)\}$$

这里 $\overline{a} \in \Delta \cup \{\bar{\varepsilon}\}$, $\overline{x}$ 是某个 $h(a)$ 的后缀.

(证明部分) 略. □

7.2.3 交, 补

CFL 在交运算下不封闭

例如, 语言 L_1 和 L_2 分别为

$$L_1 = \{0^n 1^n 2^i \mid n \geq 1, i \geq 1\}$$
$$L_2 = \{0^i 1^n 2^n \mid n \geq 1, i \geq 1\}$$

都是 CFL, 而

$$L = \{0^n 1^n 2^n \mid n \geq 1\} = L_1 \cap L_2$$

不是 CFL.

CFL 在补运算下不封闭

因为 $L_1 \cap L_2 = \overline{\overline{L_1} \cup \overline{L_2}}$, 以及 CFL 在并运算下封闭, 而在交运算下不封闭.

定理 6. *若 L 是 CFL 且 R 是正则语言, 则 $L \cap R$ 是 CFL.*

证明. 设 DFA $D = (Q_1, \Sigma, \delta_1, q_1, F_1)$ 且 $\mathbf{L}(D) = R$, PDA $P = (Q_2, \Sigma, \Gamma, \delta_2, q_2, Z_0, F_2)$ 且 $\mathbf{L}(P) = L$, 构造 PDA $P' = (Q, \Sigma, \Gamma, \delta, q_0, F)$ 如下:

(1) $Q = Q_1 \times Q_2$

(2) $q_0 = [q_1, q_2]$

(3) $F = F_1 \times F_2$

(4) δ 为

$$\delta([p, q], a, Z) = \begin{cases} \{([p, s], \beta) \mid (s, \beta) \in \delta_2(q, a, Z)\} & \text{when } a = \varepsilon \\ \{([r, s], \beta) \mid r = \delta_1(p, a) \text{ and } (s, \beta) \in \delta_2(q, a, Z)\} & \text{when } a \neq \varepsilon \end{cases}$$

那么 $\mathbf{L}(P') = L \cap R$. 证明略. □

7.2.4 封闭性的应用

语言 $L = \{ww \mid w \in (a+b)^*\}$ 不是 CFL, 可以利用封闭性和不是 CFL 的 L' 来证明. 因为

$$L \cap a^+b^+a^+b^+ = L' = \{a^ib^ja^ib^j \mid i \geq 1, j \geq 1\}$$

因为 L' 不是 CFL, 所以 L 不是 CFL.

7.3 上下文无关语言的判定性质

7.3.1 可判定的 CFL 问题

测试 CFL 的空性: 只需判断文法的开始符号 S 是否是产生的.

测试 CFL 的成员性: 利用 CNF 范式, 有 CYK 算法检查串 w 是否属于 L.

7.3.2 不可判定的 CFL 问题

与 CFL 有关的几个不可判定问题:

1. 判断给定 CFG G 的歧义性.
2. 判断给定 CFL 的固有歧义性.
3. 判断两个 CFL 的交是否为空.
4. 判断两个 CFL 是否相同.
5. 判断给定 CFL 的补是否为空. (尽管有算法判断 CFL 是否为空.)
6. 判断给定 CFL 是否等于 Σ^*.

7.4 乔姆斯基文法体系

文法 $G = (V, T, P, S)$, P 中的产生式都形如

$$\alpha \to \beta$$

其中 $\alpha \in (V \cup T)^*V(V \cup T)^*$, 即 α 中至少有一个变元, $\beta \in (V \cup T)^*$:

(1) 则 G 称为 0 型文法, 或短语结构文法 (PSG); $L(G)$ 称为 0 型语言, 短语结构语言 (PSL), 或递归可枚举语言;

(2) 若要求 $|\beta| \geq |\alpha|$, 则称 G 为 1 型文法, 或上下文有关文法 (*Context-Sensitive Language*, CSL); $L(G)$ 称为 1 型语言或上下文有关语言 (CSL);

(3) 若要求 $\alpha \in V$, 则称 G 为 2 型文法或上下文无关文法; $L(G)$ 称为 2 型语言或上下文无关语言;

(4) 若要求 $\alpha \to \beta$ 都是形如 $A \to aB$ 或 $A \to a$, 其中 $A \in V$, $a \in T$, 则称 G 是 3 型文法或正则文法; $L(G)$ 称为 3 型语言或正则语言.

乔姆斯基把文法分成这 4 种类型, 0 型文法的能力等价于图灵机, 1 型文法的能力等价于线性界限自动机. 2 型文法能力等价于非确定的下推自动机. 3 型文法也称*右线性文法*, 能力等价于有穷自动机. 文法描述语言的能力, 0 型文法最强, 3 型文法最弱.