计算理论导论

习题六: 上下文无关语言的特性

中国人民大学　信息学院　崔冠宇　2018202147

1. **2.30.** Use the pumping lemma to show that the following languages are not context free.

a. $\{0^n1^n0^n1^n | n \geq 0\}$

d. $\{t_1\#t_2\#\cdots\#t_k | k \geq 2,$ each $t_i \in \{a, b\}^*,$ and $t_i = t_j$ for some $i \neq j\}$

解:

a. 反证法。假设这是一个上下文无关语言，则设它的泵长度为 p。现在取 $w = 0^p1^p0^p1^p \in L$，$|w| \geq p$。根据泵引理，w 可被写成 $uvxyz$ 五部分，其中 $|vxy| \leq p$ 且 $|vy| > 0$，满足 $\forall i \geq 0$ 有 $uv^ixy^iz \in L$。对 vxy 进行讨论:

- 若 vxy 仅含 0，于是 vxy 完全位于某段 0 中，不妨设在第一段中。因为 $|vy| > 0$，所以 v 或 y 含有至少一个 0，于是 $uv^0xy^0z = 0^i1^p0^p1^p(i < p)$，所以 $uv^0xy^0z \notin L$;
- 若 vxy 仅含 1，与第一种情况完全类似，仍会导出 $uv^0xy^0z \notin L$;
- 若 vxy 含 0 和 1，于是
  - 若 $vxy = 0^i1^j$，此时 vxy 在第一个或第二个 0^p1^p 的交界处，此时 uv^2xy^2z 会影响相邻的两段，导致新字符串中这两段与其它两段不平衡，于是泵出的字符串不再属于这个语言。
  - 若 $vxy = 1^i0^j$，与上面的情况类似。

综上所述，无论哪种情况都会产生矛盾，故这不是上下文无关语言。

d. 反证法。假设这是一个上下文无关语言，设它的泵长度为 p。现在取 $w = a^pb^p\#a^pb^p \in L$，$|w| \geq p$。根据泵引理，w 可被写成 $uvxyz$ 五部分，其中 $|vxy| \leq p$ 且 $|vy| > 0$，满足 $\forall i \geq 0$ 有 $uv^ixy^iz \in L$。对 vxy 进行讨论:

- 若 vxy 完全在前半部分或后半部分，即 vxy 不含 #，则 $vxy = \mathtt{a}^i\mathtt{b}^j(i+j \le p)$，于是此时 uv^0xy^0z 中 $t_1 \neq t_2$，故 $uv^0xy^0z \notin L$；
- 若 vxy 跨前后两部分，即 vxy 包含 #，则 $vxy = \mathtt{a}^i\#\mathtt{b}^j(i+j < p)$，
  - 若 v 或 y 含 #，则 uv^0xy^0z 不含 #，所以 $uv^0xy^0z \notin L$；
  - 若 v 和 y 都不含 #，于是此时 $uv^0xy^0z = \mathtt{a}^p\mathtt{b}^i\#\mathtt{a}^j\mathtt{b}^p$，其中 $i \neq p$ 或 $j \neq p$，因此 $uv^0xy^0z \notin L$。

综上所述，无论哪种情况都会产生矛盾，于是这个语言不是上下文无关语言。

2. 2.31. Let B be the language of all palindromes over $\{0, 1\}$ containing equal numbers of 0s and 1s. Show that B is not context free.

证明:

反证法。假设 B 是上下文无关的，于是它满足泵引理，设泵长度为 p。取 $w = 0^p1^p1^p0^p \in B$，显然 $|w| \ge p$。根据泵引理，w 可被写成 $uvxyz$ 五部分，其中 $|vxy| \le p$ 且 $|vy| > 0$，满足 $\forall i \ge 0$ 有 $uv^ixy^iz \in L$。对 vxy 进行讨论:

- 若 vxy 仅含有 0，因为 $|vy| > 0$，于是 v 或 y 含有至少一个 0，此时 uv^0xy^0z 的 0 和 1 的数目不同，故 $uv^0xy^0z \notin L$；
- 若 vxy 仅含有 1，与上面类似，$uv^0xy^0z \notin L$；
- 若 vxy 既含有 0 也含有 1，此时 vxy 在 0 和 1 的交界处，即完全位于前半部分或完全位于后半部分，此时无论向上、向下泵都只影响原字符串前后某一段，导致新字符串不再是回文串，于是泵出的新字符串不再属于这个语言。

综上所述，无论哪种情况都会产生矛盾，于是这个语言不是上下文无关语言。

3. 2.48. Let $\Sigma = \{0, 1\}$. Let C_1 be the language of all strings that contain a 1 in their middle third. Let C_2 be the language of all strings that contain two 1s in their middle third. So

$C_1 = \{xyz | x, z \in \Sigma^* \text{ and } y \in \Sigma^* 1 \Sigma^*, \text{ where } |x| = |z| \geq |y|\}$ and

$C_2 = \{xyz | x, z \in \Sigma^* \text{ and } y \in \Sigma^* 1 \Sigma^* 1 \Sigma^*, \text{ where } |x| = |z| \geq |y|\}$.

a. Show that C_1 is a CFL.

b. Show that C_2 is not a CFL.

解:

a. 构造一个 CFG G:

- S → 0 M 0 | 0 M 1 | 1 M 0 | 1 M 1
- M → s_1 M s_2 | 1 （其中 $s_1, s_2 \in$ {0, 1, 00, 01, 10, 11}）

1. 先证明 $L(G) \subseteq C_1$。若 $S \Rightarrow^n w$（即使用第一行产生式 1 次，然后使用第二行第一类产生式 $n-2$ 次，最后使用第二行最后一个产生式 1 次），则 $w = \Sigma^i 1 \Sigma^j$（其中 $n \geq 2$，$n-1 \leq i, j \leq 2n-3$），
   - 若 $i \leq j$，取 $x = \Sigma^i, y = 1\Sigma^{j-i}, z = \Sigma^i$，于是
     $|y| = j - i + 1 \leq (2n-3) - (n-1) + 1 = n - 1 \leq i = |x| = |z|$;
   - 若 $i > j$，与上面类似。

   于是 $\forall x \in L(G)$ 都有 $x \in C_1$，即 $L(G) \subseteq C_1$。
2. 再证明 $C_1 \subseteq L(G)$。从 C_1 中任取 s，
   - 若 $|s| = 3$，显然 s 为 `010`，`011`，`110`，`111` 之一，显然 $s \in L(G)$;
   - 由 C_1 的定义，$|s| = 4$ 是不可能的;
   - 若 $|s| = n \geq 5$，$s = \Sigma^a 1 \Sigma^b$，满足 $a + b = n - 1$，$a \leq 2b - 1$ 以及 $b \leq 2a - 1$。不妨设 $a < b$，容易看出 s 能被 G 产生（每次只需选择长度为 1 的 s_1，适当选择 s_2 即可）。

   于是 $\forall x \in C_1$ 都有 $x \in L(G)$，即 $C_1 \subseteq L(G)$。

综上，$C_1 = L(G)$，于是 C_1 是上下文无关的。

b. C_2 不是上下文无关语言，用反证法。假设 C_2 是一个上下文无关语言，则设它的泵长度为 p。现在取

$w = \texttt{0}^{p+2}\texttt{1}\texttt{0}^{p}\texttt{1}\texttt{0}^{p+2} \in L$，$|w| \geq p$。根据泵引理，$w$ 可被写成 $uvxyz$ 五部分，其中 $|vxy| \leq p$ 且 $|vy| > 0$，满足 $\forall i \geq 0$ 有 $uv^ixy^iz \in L$。对 vy 进行讨论:

- 若 $vy \neq \texttt{0}^a$，说明 vy 中有 1，uv^0xy^0z 仅含有一个 1，$uxz \notin C_2$；
- 若 $vy = \texttt{0}^a$，且
  - vxy 完全来自于第一段或第三段 0，于是 uv^0xy^0z 为 $\texttt{0}^{p+2-a}\texttt{1}\texttt{0}^{p}\texttt{1}\texttt{0}^{p+2}$ 或 $\texttt{0}^{p+2}\texttt{1}\texttt{0}^{p}\texttt{1}\texttt{0}^{p+2-a}$，都不属于 C_2；
  - vxy 完全来自于第二段 0，于是 $uv^2xy^2z = \texttt{0}^{p+2}\texttt{1}\texttt{0}^{p+a}\texttt{1}\texttt{0}^{p+2} \notin C_2$；
  - $x = \texttt{1}$，不妨设它是第一个 1（另一种情况对称），此时 $v = \texttt{0}^b$、$v = \texttt{0}^c$，$uv^2xy^2z = \texttt{0}^{p+2+b}\texttt{1}\texttt{0}^{p+c}\texttt{1}\texttt{0}^{p+2}$，第二个 1 被挤到第三段，于是 $uv^2xy^2z \notin C_2$。

综上所述，无论哪种情况都会产生矛盾，于是这个语言不是上下文无关语言。

4. Consider the following operation on languages:

$$\text{INIT}(L) = \{x | \text{ for some } y, \text{ the string } xy \text{ is in } L\}.$$

Prove that context-free languages are closed under the INIT operation. (Remarks: There are several ways to establish such results: a PDA construction, a grammar construction, or using known closure properties. We ask that you describe your construction in complete detail but require only a brief justification of the correctness of your construction.)

思路: 类似于正则语言中的 SKIP 闭包，可以考虑对 PDA 进行改造，以模拟 INIT 闭包的行为。

证明:

因为 L 是上下文无关的，因此有 PDA $M = (Q_M, \Sigma, \Gamma, \delta_M, q_M, F_M)$ 识别语言 L。下面构造一台 PDA $N = (Q_N, \Sigma, \Gamma, \delta_N, q_N, F_N)$（以空栈方式）识别 INIT($L$)：其中，

- $Q_N = Q_M \cup Q'_M = Q_M \cup \{q' | q \in Q_M\}$，即拷贝一份 M 的状态放在一边。

- δ_N 如下所定义:

$$\delta_N(p,\sigma,\gamma)=\begin{cases}\delta_M(q,\sigma,\gamma) & p=q\in Q_M,\sigma\neq\varepsilon \text{ 或 } \gamma\neq\varepsilon\\ \delta_M(q,\varepsilon,\varepsilon)\cup\{(q',\varepsilon)\} & p=q\in Q_M,\sigma=\varepsilon \text{ 且 } \gamma=\varepsilon\\ [\delta_M(q,\varepsilon,\gamma)]' & p=q'\in Q_M',\sigma=\varepsilon\\ \emptyset & \text{其它}\end{cases}$$

  即 N 保留了 M 原有状态间的转移，在原有状态与对应复制状态间增加转移 $\varepsilon,\varepsilon\to\varepsilon$，并且把原有状态之间的转移 $a,b\to c$ 变为对应复制状态间的转移 $\varepsilon,b\to c$。

- $q_N=q_M$，即 N 的新起始状态就是原起始状态。

- $F_N=F_M\cup F_M'=F_M\cup\{q'|q\in F\}$，即新接受状态是原接受状态与对应复制状态的并集。

正确性的简单说明:

1. $x\in \text{INIT}(L)\Rightarrow x\in L(N)$，即 $\text{INIT}(L)\subseteq L(N)$。设 $x\in\text{INIT}(L)$，根据定义，存在 $w=xy=w_0w_1\cdots w_{n-1}$（可能有 ε 占位）使得 $w\in L$。于是对于 w 有计算序列 $\{(q_i,s_i)\}_{i=0}^n$ 满足

   - $q_0=q_M=q_N$，$s_0=\varepsilon$（起始状态、空栈）;
   - $\forall 1\leq i\leq n,(q_i,b)\in\delta(q_{i-1},w_{i-1},a)$（对某 $a,b\in\Gamma_\varepsilon$ 有 $s_i=at$ 且 $s_{i+1}=bt$）;
   - $q_n\in F_M\subseteq F_N,s_n=\varepsilon$（以空栈终止在接受态上）。

   根据 N 的定义，这些序列可以“移植”到 N 上。现在将前缀 x 在 N 上按上述序列运行，设它停在某个 (q_i,s_i) 上，接下来它可以通过 $\varepsilon,\varepsilon\to\varepsilon$ 转移到 (q_i',s_i)，然后在读取 ε 的同时，模拟 w 对栈进行后续操作，直到 $q_n'\in F_N$。根据 N 的定义，x 的一个计算序列为

   $(q_0=q_N,s_0=\varepsilon),(q_1,s_1),\cdots,(q_i,s_i),(q_i',s_i),(q_{i+1}',s_{i+1}),\cdots,(q_n',s_n=\varepsilon)$。

   因此，每个前缀 $x\in\text{INIT}(L)$ 都会被 N 接受，于是 $\text{INIT}(L)\subseteq L(N)$。

2. $x \in L(N) \Rightarrow x \in \text{INIT}(L)$，即 $L(N) \subseteq \text{INIT}(L)$。设 $x \in L(N)$，根据定义，存在 x 的计算序列 $\{(q_i, s_i)\}_{i=0}^{n}$ 满足上述三点。对 q_i 进行讨论:

   - 各 $q_i \in Q_M$，显然 $x \in L(M) = L \subseteq \text{INIT}(L)$，因此 $x \in \text{INIT}(L)$；
   - 若某 $q_i \in Q'_M$（取 i 最小），根据 N 的构造，对任意 $j \geq i$ 一定有 $q_j \in Q'_M$。此时很容易找到某 $w \in L$ 被 N 接受，且 w 的计算序列 $\{(r_i, s_i)\}$ 满足
     - 对于 $j < i$，$r_j = q_j$，对于 $j \geq i$，$r'_j = q_j$。
     - x 是 w 的前缀。

     此时，也有 $x \in \text{INIT}(L)$。

综上，$\text{INIT}(L) = L(N)$，因此 $\text{INIT}(L)$ 是上下文无关的。由此知上下文无关语言对 INIT 运算封闭。

5. Given the grammar G as the following:

   - S → AB | BC
   - A → BA | a
   - B → CC | b
   - C → AB | a

Use the CYK algorithm to determine the membership of the string ababa in $L(G)$.

解:

运行 CYK 算法结果如下:

S, A				
B	B			
B	S, C	B		
S, C	S, A	S, C	S, A	
A, C	B	A, C	B	A, C
a	b	a	b	a

可见，S $\in X_{1,5}$，因此 ababa$\in L(G)$。