

形式语言与自动机理论

不可判定性

王春宇

chunyu@hit.edu.cn

计算学部

哈尔滨工业大学

2021 年 4 月

不可判定性

- 不可判定性
- 非递归可枚举的语言
- 递归可枚举但非递归的语言
- 语言类的关系



不可判定问题

定义

一个问题, 如果它的语言是递归的, 称为**可判定问题**, 否则称为**不可判定问题**.

- 递归可枚举语言 — 图灵机所识别
- 递归语言 — 保证停机的图灵机所识别

不可判定的问题

- 不存在保证停机的图灵机识别该问题的语言
- 不存在解决该问题的算法

不可判定性

- 不可判定性
- 非递归可枚举的语言
 - 第 i 个串
 - 图灵机编码与第 i 个图灵机
 - 对角化语言 L_d
- 递归可枚举但非递归的语言
- 语言类的关系



判定问题

“图灵机 M 接受输入 w 吗?”

第 i 个串 w_i

定义

将全部 $(0+1)^*$ 中的字符串按长度和字典序排序, 那么第 i 个串就是 w_i .
且刚好有

$$\text{binary}(i) = 1w_i.$$

即:

i	1	2	3	4	5	6	7	8	9	...
$\text{binary}(i)$	1ε	10	11	100	101	110	111	1000	1001	...
w_i	ε	0	1	00	01	10	11	000	001	...

图灵机编码

将 $\Sigma = \{0, 1\}$ 上的任意图灵机 M , 用二进制字符串编码

$$M = (Q, \Sigma, \Gamma, \delta, q_1, B, F)$$

- ① $Q = \{q_1, q_2, \dots, q_{|Q|}\}$, 开始状态为 q_1 , 终态为 q_2 且停机;
- ② $\Gamma = \{X_1, X_2, \dots, X_{|\Gamma|}\}$, 总有 $X_1 = 0, X_2 = 1, X_3 = B$;
- ③ 设带头移动方向 $D_1 = L, D_2 = R$;
- ④ 任意的转移 $\delta(q_i, X_j) = (q_k, X_l, D_m)$ 编码为

$$C = 0^i 10^j 10^k 10^l 10^m;$$

- ⑤ 则全部 n 个转移的编码合并在一起, 作为图灵机 M 的编码:

$$C_1 11 C_2 11 \dots C_{n-1} 11 C_n.$$

第 i 个图灵机 M_i

定义

如果图灵机 M 的编码为第 i 个串 w_i , 则称 M 是第 i 个图灵机 M_i .

- 任意图灵机 M , 都对应一个字符串 w ;
- 任意字符串 w , 也都可以看作一个图灵机的编码;
- 如果编码不合法, 则将其看作接受 \emptyset 且立即停机的图灵机.

非递归可枚举的语言

定义

使第 i 个串 w_i 不属于第 i 个图灵机 M_i 的语言 $\mathbf{L}(M_i)$ 的所有 w_i 的集合 L_d , 称为**对角化语言**

$$L_d = \{ w_i \mid w_i \notin \mathbf{L}(M_i), i \geq 1 \}.$$

		$w_j \longrightarrow$					
		1	2	3	4	5	\cdots
M_i \downarrow	1	0	0	1	1	0	\cdots
	2	1	0	0	1	0	\cdots
	3	0	1	1	0	0	\cdots
	4	0	0	1	1	1	\cdots
	5	1	1	0	0	0	\cdots
	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\ddots

定理 45

L_d 不是递归可枚举语言, 即不存在图灵机接受 L_d .

定理 45

L_d 不是递归可枚举语言, 即不存在图灵机接受 L_d .

证明: (反证法) 假设存在识别 L_d 的图灵机 M , 那么 M 也可被编码, 不妨设它是第 i 个图灵机 $M_i = M$, 即 $\mathbf{L}(M_i) = L_d$.

那么, 考虑第 i 个串 w_i 是否会被 M_i 识别:

- ① 如果 $w_i \in \mathbf{L}(M_i) = L_d$, 那么由 L_d 的定义, 又有 $w_i \notin \mathbf{L}(M_i)$;
- ② 如果 $w_i \notin \mathbf{L}(M_i)$, 那么由 L_d 的定义, 又有 $w_i \in L_d = \mathbf{L}(M_i)$.

无论如何都会矛盾, 因此假设不成立, 不存在接受 L_d 的图灵机.



不可判定性

- 不可判定性
- 非递归可枚举的语言
- 递归可枚举但非递归的语言
 - 递归语言的封闭性
 - 通用语言与通用图灵机
- 语言类的关系

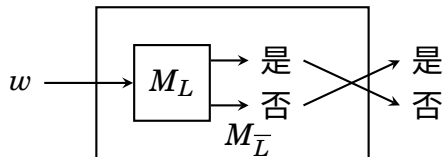


递归语言的封闭性

定理 46

如果 L 是递归的, 那么 \bar{L} 也是递归的.

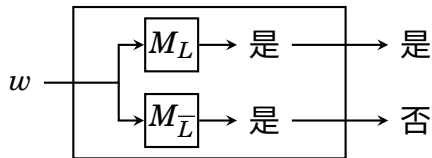
证明:



定理 47

如果语言 L 和 \bar{L} 都是递归可枚举的, 那么 L 是递归的.

证明:



语言和它的补之间的关系

如果 L 和 \bar{L} 是一对互补的语言, 那么:

- ① 两者都是递归的, 或者
- ② 两者都不是递归可枚举的, 或者
- ③ 其中之一是递归可枚举的但不是递归的, 而另一个不是递归可枚举的.

通用语言

定义

图灵机 M 和输入串 w 组成的有序对 (M, w) , 可编码为 $M111w$.

定义

如果图灵机 M 接受串 w , 那么有序对 (M, w) 构成的语言 L_u , 称为通用语言

$$L_u = \{ M111w \mid w \in \mathbf{L}(M) \}.$$

定理 48

通用语言 L_u 不是递归的.

证明: 假设存在算法 A 识别 L_u , 则可构造识别 L_d 的算法 B .

将 B 的输入 $w = w_i$ 转换为 $M_i 111 w_i$ 交给 A 判断:

- 当 A 接受, 表示 $w_i \in \mathbf{L}(M_i)$, 则 B 拒绝;
- 当 A 拒绝, 表示 $w_i \notin \mathbf{L}(M_i)$, 则 B 接受.

而由于 L_d 不是递归的, 所以 B 不可能存在, 所以 L_u 不可能是递归的. □

通用图灵机

定理 49

通用语言 L_u 是递归可枚举的.

证明:

构造图灵机 U , 当输入 $M111w$ 时, 用 3 条带模拟 M 处理串 w 的过程:

- ① 第 1 带装载 M 的编码;
- ② 第 2 带模拟 M 的带, 放置串 w ;
- ③ 第 3 带存储 M 的状态.

因为 M 接受 w 时会停机, 因此 U 可以识别 L_u . □

定义

可以模拟其他任意图灵机 M 的图灵机 U , 称为通用图灵机.

通用图灵机的重要意义

- 识别 L_u 的通用图灵机 U , 可以模拟任意图灵机;
- 冯·诺伊曼通用数字电子计算机体系结构设计思想的灵感来源.
- 抽象理论的先期发展可以对实际问题有很大帮助.

通用图灵机的编号

- 罗杰·彭罗斯在《皇帝新脑》中以另一种编码给出了 $U = M_i$ 的 i 为:

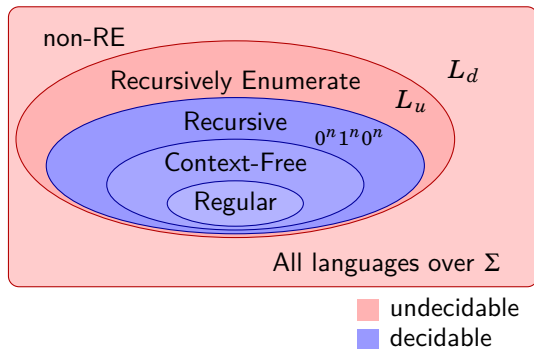
724485533533931757719839503961571123795236067255655963110814479660650505940424109031048361363
235936564444345838222688327876762655614469281411771501784255170755408565768975334635694247848
859704693472573998858228382779529468346052106116983594593879188554632644092552550582055598945
189071653741489603309675302043155362503498452983232065158304766414213070881932971723415105698
026273468642992183817215733348282307345371342147505974034518437235959309064002432107734217885
149276079759763441512307958639635449226915947965461471134570014504816733756217257346452273105
448298078496512698878896456976090663420447798902191443793283001949357096392170390483327088259
62013017737272027186259199144282754374223513556751340842229988937441053430547104436869587640
517812801943753081387063994277282315642528923751456544389905278079324114482614235728619311833
261065612275553181020751108533763380603108236167504563585216421486954234718742643754442879006
248582709124042207653875426445413345174856629157429990950262300973373813772416217274772361020
678685400289356608569682262014198248621698902609130940298570600174300670086896759034473417412
787425581201549366393899690581773859165405535670409282133222163141097871081459978669599704509
681841906299443656015145490488092208448003482249207730403043188429899393135266882349662101947
161910701461968523192847482034495897709553561107027581748733327296678998798473284098190764851
272631001740166787363477605857245036964434897992034489997455662402937487668839751404451665707
750060513883991668814072545544665222050724262392379211525318162512536305093172863142200406457
1305275802307665183351995689139748137504926429605010013651980186945639498 (1654 位)

不可判定性

- 不可判定性
- 非递归可枚举的语言
- 递归可枚举但非递归的语言
- 语言类的关系



语言类的关系





哈爾濱工業大學

HARBIN INSTITUTE OF TECHNOLOGY

chunyu@hit.edu.cn
<http://nclab.net/~chunyu>

