



正则语言与正则文法

姚刚

目录

正则表达式

正则表达式和正则语言

正则文法

第三章 正则语言与正则文法

姚 刚

中国科学院信息工程研究所



正则表达式表示语言(续)

正则语言与正则文法

姚刚

目录

正则表达式

正则表达式和正则语言

正则文法

定义

如果 r_1 和 r_2 是正则表达式, 那么

- $L(r_1 + r_2) = L(r_1) \cup L(r_2)$ 。
- $L(r_1 \cdot r_2) = L(r_1)L(r_2)$ 。
- $L((r_1)) = L(r_1)$ 。
- $L(r_1^*) = (L(r_1))^*$ 。

语言的并

这样就可以给出正则表达式对应的语言



优先级

正则语言与正则文法

姚刚

目录

正则表达式

正则表达式和正则语言

正则文法

对于一个复杂的表达式，可能有多种拆分方法。为了避免这种拆分上的歧义，一种解决办法是使用圆括号将表达式扩起来，但是这样得到的结果又过于麻烦。于是，我们使用数学和程序语言中类似的规定，建立一套优先级规则：**星闭包要优先于连接，连接要优先于并。**

另外，连接符号通常省略，即使用 r_1r_2 代替 $r_1 \cdot r_2$ 。

星 > 乘法 > 加法 * . +



例子

正则语言与正则文法

姚刚

目录

正则表达式

正则表达式和正则语言

正则文法

设 $\Sigma = \{a, b\}$, 则表达式 $(a+b)^*(a+bb)$ 是正则的, 给出其表达的语言。

表达式 $(aa)^*(bb)^*b$ 也是正则的, 给出其表达的语言。

任意符号串 $(a+b)^*$

设 $\Sigma = \{0, 1\}$, 给出满足描述 $L(r) = \{w \in \Sigma^* : w \text{ 至少有一对连续的 } 0\}$ 的正则表达式。

$(1+01)^*(0+\varepsilon)$

设 $\Sigma = \{0, 1\}$, 给出 $L(r) = \{w \in \Sigma^* : w \text{ 中不存在连续的 } 0\}$ 的正则表达式。

两个语言互补



定理

第一步：基本正则表达式

第二步： $r_1 r_2$ 用NDF表示

第三步：运算 r_1+r_2 r_1r_2

正则语言与正则文法

姚刚

目录

正则表达式

正则表达式和正则语言

正则文法

定理

设 r 是正则表达式，那么，一定存在某个非确定型有穷接受器接受 $L(r)$ 。因此， $L(r)$ 是正则语言。

构造一个NFA，使其接受语言 $L(r)$ ，其中

$$r = (a + bb)^*(ba^* + \varepsilon)。$$



例子

正则语言与正则文法

姚刚

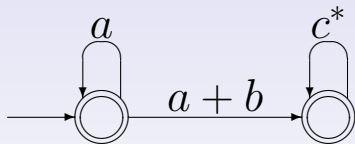
目录

正则表达式

正则表达式和正则语言

正则文法

对于一个通用转移图，考察他接受的语言。



输入a或b都转移到后面



正则语言与正则文法

姚刚

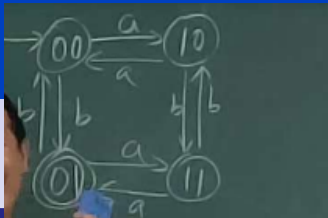
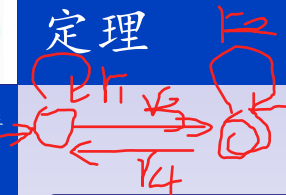
目录

正则表达式

正则表达式和正则语言

正则文法

定理



定理

拿到正则语言，一定能找到正则表达式

设 L 是正则语言，那么总存在正则表达式，使得 $L = L(r)$ 。

例子很有趣 四状态奇偶性

构造语言 L 对应的正则表达式。其中 $L = \{w \in \{a, b\}^* : n_a(w) \text{ 是偶数, 并且 } n_b(w) \text{ 是奇数.} \}$

第一步 假设只有一个终止状态 并且和初始状态不是同一个 多个 通过空转移

去掉中间状态 并且去掉后 边上能够用正则表达式表示

慢慢去掉中间状态 得到画的很丑的那个图 $r_1 r_2 r_3 r_4$



例子

正则语言与正则文法

姚刚

目录

正则表达式

正则表达式和正则语言

正则文法

- 文法 $G_1 = (\{S\}, \{a, b\}, S, P_1)$, 其中 P_1 定义为 $S \rightarrow abS|a$ 。
- 文法 $G_2 = (\{S, S_1, S_2\}, \{a, b\}, S, P_2)$, 其中 P_2 定义为

$$S \rightarrow S_1ab \quad S_1 \rightarrow S_1ab|S_2 \quad S_2 \rightarrow a。$$

- 文法 $G = (\{S, A, B\}, \{a, b\}, S, P)$, 其中 P 定义为

$$S \rightarrow A \quad A \rightarrow aB|\varepsilon \quad B \rightarrow Ab。$$