第五章 模式识别中的句法分析方法

中科大 自动化系 郑志刚

2018.11

- 5.1概述
- 5.2形式语言理论基础
- 5.3自动机理论
- 5.4基元提取

§5.1 概述

- 统计模式识别是基于模式特征的一组测量值来组成特征向量，用决策理论划分特征空间的方法进行分类。
- 基于描述模式的结构信息，用形式语言中的规则进行分类，更典型地应用于景物图片的分析。
- 因为在这类问题中，所研究的模式通常十分复杂，需要的特征也很多，仅用数值上的特征不足以反映它们的类别。

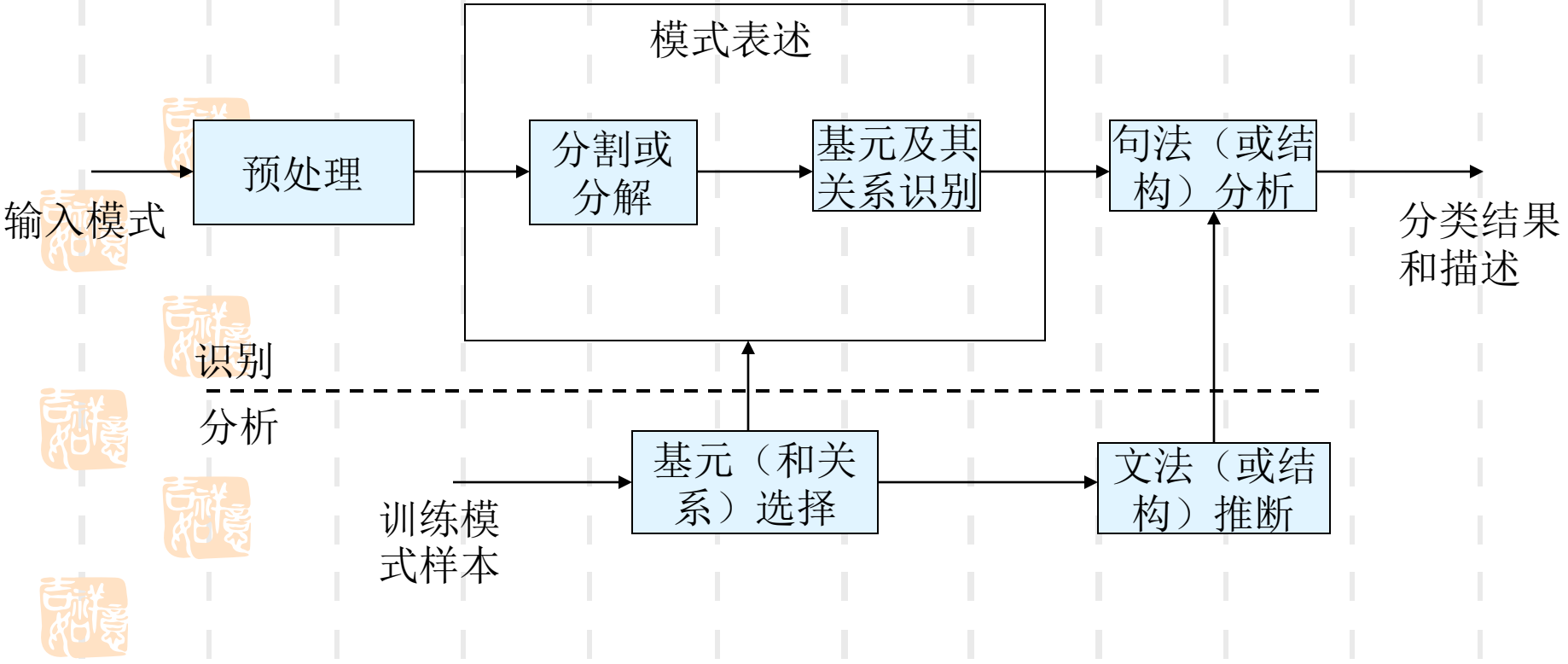
- 结构信息重要，如图片，语音。景物的识别十分复杂，要求特征量非常巨大，要把每一模式分类准确很困难，希望把一个

$$\begin{cases} \text{复杂模式}--\sum\text{若干简单子模式模式组合} \\ \text{子模式}--\sum\text{若干基元（源模式）} \end{cases}$$

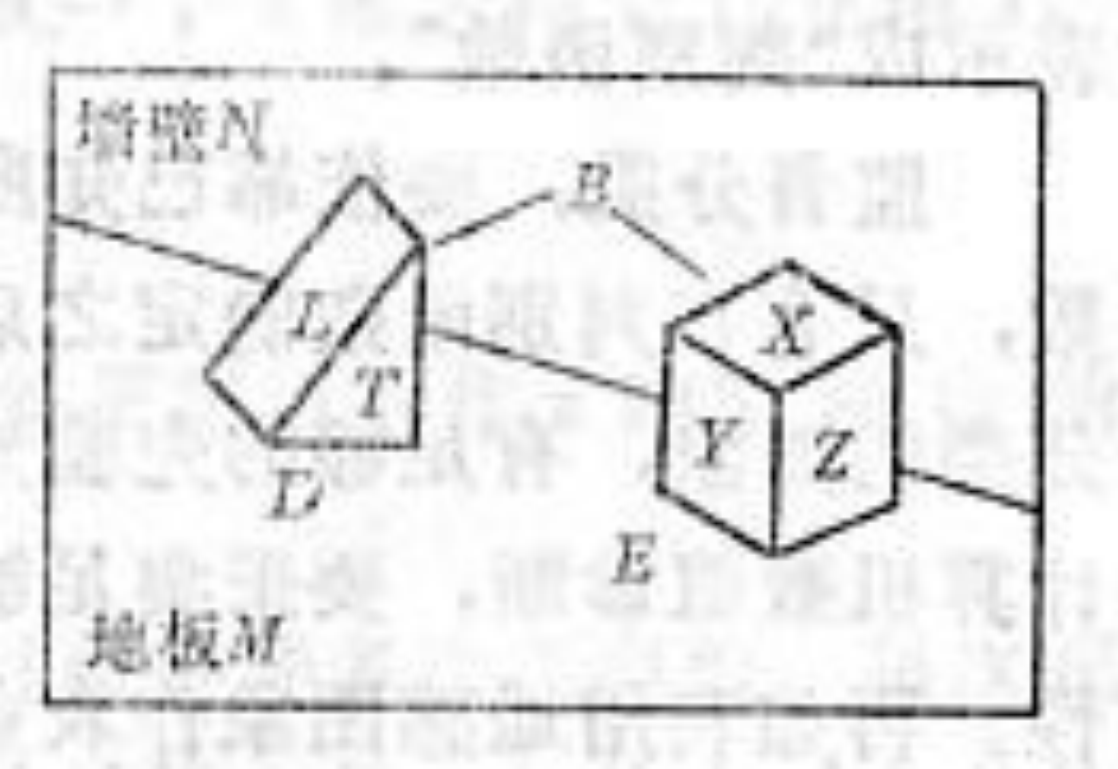
- 识别基元⟶子模式⟶复杂模式 （汉字，指纹，连续语音采用这方法已获得一定成功）

- 句法模式识别系统的组成
  - 图像预处理
  - 图像分割
  - 基元及其关系识别
  - 句法分析

- 句法模式识别系统框图


模式表述
预处理
分割或分解
基元及其关系识别
句法（或结构）分析
输入模式
分类结果和描述
识别分析
基元（和关系）选择
文法（或结构）推断
训练模式样本

- 对图像的结构信息描述



- 句法模式识别系统处理过程
  - 待识别的输入图像，经过增强、去噪声等处理后，按识别的具体对象分割成子图；
    - 三角体D和长方体E
  - 然后将子图分割成更简单的模式基元；
    - 组成三角体和长方体的各个面{L,T}和{X,Y,Z}
  - 判别基元之间的关系。
    - 三角体D是由相互邻接的四边形L和三角形T组成
    - 长方体E是有三个相互邻接的四边形X,Y和Z组成

- 句法模式识别系统处理过程
  - 基元本身包含的结构信息已不多，仅需少量特征即可识别。
  - 如果用有限个字符代表不同的基元，则由基元按一定结构关系组成的子图或图形可以用一个有序的字符串来代表。
  - 假如事先用形式语言的规则从字符串中推断出能生成它的文法，则可以通过句法分析，按给定的句法（文法）来辨识由基元字符组成的句子，从而判别它是否属于由该给定文法所能描述的模式类，达到分类的目的

- 句法模式识别学习过程
  - 为了要事先确定一个文法来描述所要研究模式的结构信息，同样需要采用模式的训练样本集把文法推断出来。
  - 有了推断出来的文法，才可以对未知类别的字符串进行句法分析，达到分类的目的。
  - 这一过程类似于统计模式识别中的学习过程，但文法推断过程远不及统计学习来的成熟。

§5.2 形式语言理论基础

- 1.集合的基础知识
- 2.关系
- 3.语言
- 4.文法

1.集合的基础知识

1.1 集合及其表示

1.2 集合之间的关系

1.3 集合的运算

1.1 集合及其表示

- 一些没有重复的对象的全体称为集合(set)，而这些被包含的对象称为该集合的元素(element)。集合中元素可以按任意的顺序进行排列。一般，使用大写英文字母表示一个集合。

列举法

- 对于元素个数较少的集合，可以采用列举法，即将集合的所有元素全部列出，并放在一对花括号中。例如集合A ={0，1，2，3，4，5，6，7，8，9}

命题法

- 对于集合元素较多的或者是由无穷多个元素组成的集合，可以使用集合形成模式$\{x \mid P(x)\}$进行描述，其中，x表示集合中的任一元素，$P(x)$是一个谓词，对x进行限定，$\{x \mid P(x)\}$表示由满足$P(x)$的一切x构成的集合。可以使用自然语言，或者数学表示法来描述谓词$P(x)$。
- 例如，$\{n \mid n \bmod 2 = 0\}$，表明了一个由所有偶数组成的集合。

集合的基数

- 如果集合A包含元素x（也称元素x在集合A中），记为$x \in A$。
- 对于任意的有穷集合A，使用$|A|$表示该集合包含的元素的个数，也称基数或势。显然，$|A| = 0 \Leftrightarrow A = \varnothing$。
- 如果一个集合中的元素个数是有限的，称该集合为有穷集合。如果一个集合包含的元素是无限的，称该集合为无穷集合。无穷集合又分为可数集(如自然数集，有理数集)和不可数集(如实数集)。

1.2 集合之间的关系

- 定义1-2 设A, B是两个集合，如果集合A中的每个元素都是集合B的元素，则称集合A是集合B的子集，集合B是集合A的包集。记作$A \subseteq B$，或$B \supseteq A$。
- 定义1-3 设A, B是两个集合，如果$A \subseteq B$，且$\exists x \in B$，但$x \notin A$，则称A是B的真子集，记作$A \subset B$。

几个结论

- $A = B$ iff $A \subseteq B$且$B \subseteq A$。
- 如果A是有穷集，且$A \subset B$，则$|A| < |B|$。
- 对于无穷集，这个结论并不适用。

1.3 集合的运算

- 并 **A ∪ B**
- 交**A∩B**
- 差**A-B**

笛卡儿乘积的定义

- 集合A和B的笛卡儿乘积使用A×B表示（也简记为AB），它是集合

  $\{(a,b) \mid a \in A \text{ 且 } b \in B\}$。

- A×B的元素称为有序偶对(a,b)，总是A的元素在前，B的元素在后。
- A×B与B×A一般不相等。
- 例：设A={a，b，c},B={0，1}；
- 则

A×B={(a,0),(a,1),(b,0),(b,1),(c,0),(c,1)}

- 而

B×A={(0,a),(0,b),(0,c),(1,a),(1,b),(1,c)}

幂集

- 设A为一个集合，那么A的幂集为A的所有子集组成的集合，记为2^A，即$2^A=\{B \mid B \subseteq A\}$。
- 例如，集合$A=\{1，2，3\}$，则A的幂集为：

  $2^A=\{Ø，\{1\}，\{2\}，\{3\}，\{1，2\}，\{1，3\}，\{2，3\}，\{1，2，3\}\}$。
- 当集合A为有穷集时，如果集合A包含的元素个数为n，那么集合2^A的元素个数(集合A的所有子集的个数)为2^n，这就是称2^A为集合A的幂集的原因。

2 关系

- 2.1 二元关系
- 2.2 等价关系
- 2.3 关系的合成
- 2.4 关系的闭包

2.1 二元关系

- 定义：设A,B是两个集合，任意的$R \subseteq A \times B$，R是A到B的二元关系(binary relation)。A称为定义域(domain)，B称为值域(range)。当A=B时，称R是A上的二元关系。
- $(a,b) \in R$表示a与b满足关系R，按照中缀形式，也可表示为aRb。

2.1 二元关系

例：集合{1,3,4,8}和{0,3,5,7}的元素之间存在的小于关系有R<={(1,3), (1,5), (1,7), (3,5), (3,7), (4,5), (4,7)}，大于关系有R>={(1,0), (3,0), (4,0), (4,3), (8,0), (8,3), (8,5),(8,7)}，显然$R_<$,$R_>$ ⊆{1,3,4,8}×{0,3,5,7}。

写成中缀形式为1<3…，1>0…。

关系

设R是A上的二元关系，有

- (1) 如果对$\forall a \in A$，都有$(a, a) \in R$，则称R是自反的。
- (2) 如果对$\forall a \in A$，都有$(a, a) \notin R$，则称R是反自反的。
- (3) 如果对$\forall a, b \in A$，$(a, b) \in R \Rightarrow (b, a) \in R$，则称$R$是对称的。

关系

- **(4)** 如果对$\forall a, b \in A$，$(a, b) \in R$且$(b, a) \in R \Rightarrow a = b$，则称**R**是反对称的。
- **(5)** 如果对$\forall a, b, c \in A$，$(a, b) \in R$且$(b, c) \in R \Rightarrow (a, c) \in R$，则称**R**为传递的。

例：(1)“=”关系是自反的、对称的、传递的。

(2)“>”、“<”关系是反自反的、传递的。

(3)“≥”、“≤”关系是自反的、反对称的、传递的。

(4)集合之间的包含关系是自反的、反对称的、传递的。

2.2 等价关系

- 定义 如果集合***A***上的二元关系***R***是自反的、对称的和传递的，则称***R***为等价关系。

2.3 关系的合成

- 定义 设$R_1 \subseteq A \times B$是A到B的关系，$R_2 \subseteq B \times C$是B到C的关系，则R_1与R_2的合成是A到C的关系

$$R_1 R_2 = \{(a, c) | \exists (a, b) \in R_1 \text{且} (b, c) \in R_2\}$$

例：设R_1,R_2是集合{1,2,3,4}上的关系，其中

R_1={(1,1),(1,2),(2,3),(3,4)},

R_2={(2,4),(4,1),(4,3),(3,1),(3,4)}

则$R_1 \circ R_2$={(1,4),(2,1),(2,4),(3,1),(3,3)}。

关系的n次幂

- 定义1-17 设R是S上的二元关系，则R^n如下递归定义：
- (1) $R^0 = \{(a, a) \mid a \in S\}$
- (2) $R^1 = R$
- (3) $R^n = R^{n-1} R$ $(n = 2, 3, \ldots)$

2.4 关系的闭包

- 定义 设R是S上的二元关系，R的正闭包R^+定义为

(1) $R \subseteq R^+$

(2) 如果$(a, b), (b, c) \in R^+$, 则$(a, c) \in R^+$

(3) 除(1)，(2)外，R^+不再含有其他任何元素。

$R^+ = R \cup R^2 \cup R^3 \cup \ldots$

且当S为有穷集时，有

$R^+ = R \cup R^2 \cup R^3 \cup \ldots R^{|S|}$

关系的克林闭包

$$R^* = R^0 \cup R^+$$

即

- $R^* = R^0 \cup R^1 \cup R^2 \cup \ldots \cup R^n \cup \ldots$
- $R^+ = R^1 \cup R^2 \cup \ldots \cup R^n \cup \ldots$
- 当S为有穷集合时，
- $R^* = R^0 \cup R^1 \cup R^2 \cup \ldots \cup R^{|s|}$
- $R^+ = R^1 \cup R^2 \cup \ldots \cup R^{|s|}$

3 语言

- 3.1 什么是语言
- 3.2 基本概念

3.1 什么是语言

- 语言是“为相当大的团体的人所懂得并使用的字和组合这些字的方法的统一”，但这个定义对于计算机界的人们来说毫无意义。

- 语言是一组规定（或称为规则）的组合。

- 字母表
- 单词符号的形成规则
- 单词符号的含义规则
- 语法单位的形成规则（语法规则）
- 语法单位的含义规则（语义规则）

3.2 基本概念

- 对语言研究的几个方面
  - 语言的表示
  - 给定语言是否存在有穷描述
  - 有穷表示的语言的结构

一、语言的一些术语：

- 字母表： 字符的有限集合，记为T。
- 字符串： 由字母表T中的字符构成的序列称字母表T上的字符串（句子）。
  - 常记为u,v,w,x,y,z；
  - 常用a,b,c,d 标识单个字符。

字 母 表 （*Alphabet*）

- 概念　形式符号的集合
- 记号　常用 T、Σ 表示
- 举例
  - 英文字母表 { *a*, *b*, …, *z*, *A*, *B* , …, *Z* }
  - 英文标点符号表 { , ; : . ? ! ’ ‘ “ ” ( ) … }
  - 汉字表 { …, 自, …, 动 , …, 机, … }
  - 化学元素表 { H, He, Li, …, }
  - T = { *a*, *n*, *y*, 任,意 }

字 符 串 （*string*）

✧ 概念　字母表 T 上的一个字符串（简称串），或称为字（*word*），为 T 中字符构成的一个有限序列。空串（*empty string*），用 ε 表示，不包含任何字符。

举例　设 T = { *a*, *b* }，则 ε, *a*, *ba*, *bbaba* 等都是串

✧ 字符串 *w* 的长度，记为 $|w|$，是包含在 *w* 中字符的个数

举例　$|\varepsilon| = 0$，　$|bbaba| = 5$

a^i 表示含有i个a的字符串

关 于 字 符 串 的 运 算

- 连接（*concatenation*）

  设 x, y为串，且 $x = a_1a_2 \ldots a_m$, $y = b_1b_2 \ldots b_n$,

  则 x 与 y 的连接

  $$x\,y = a_1a_2 \ldots a_m\,b_1b_2 \ldots b_n$$

- 连接运算的性质
  - $(x\,y)\,z = x\,(y\,z)$
  - $\varepsilon x = x\,\varepsilon = x$
  - $|x\,y| = |x| + |y|$

关 于 字 符 串 的 运 算

- 其它　如 取头字符，取尾部，子串匹配 等
  - 设ω1, ω2, ω3是字母表T上的字符串，称ω1是字符串ω1ω2的前缀，ω2是字符串ω1ω2的后缀，且ω2是字符串ω1ω2ω3的子串。
  - 空串是任何字符串的前缀，后缀及子串。
  - 例：

    abc的前缀　a　ab　abc ε.

    后缀　c　bc　abc ε.

    子串　a　b　c　ab　bc　abc　ε,

    即一个字符串可以看作是多个字符串的连接。

- 字符串ω的逆用 ϖ 表示。 是字符串ω的倒置。

  $\omega = b_1 b_2 \cdots\cdots b_n$

  $\varpi = b_n b_{n-1} \cdots\cdots b_2 b_1$

- 空串ε的逆还是ε

字 母 表 的 幂 运 算

✧ 幂运算　设 T 为字母表，n 为任意自然数，

定义（1）$T^0 = \{ \varepsilon \}$

（2）设 $x \in T^{n-1}$，$a \in T$，则 $ax \in T^n$

（3）T^n 中的元素只能由（1）和（2）生成

✧ $*$ 闭包　$T^* = T^0 \cup T^1 \cup T^2 \cup \ldots$

✧ $+$ 闭包　$T^+ = T^1 \cup T^2 \cup T^3 \cup \ldots$

✧ $T^* = T^+ \cup \{ \varepsilon \}$，　$T^+ = T^* - \{ \varepsilon \}$

闭包的物理意义

✧ **T的星号闭包T*：**字母表T上的所有字符串和空串的集合。

✧ **T的正闭包T+：**字母表T上的所有字符串构成的集合。

$$T^* = T^+ \cup \{\varepsilon\}$$

✧**举例**　　设 T = { 0，1 }，则

$T^0 = \{\varepsilon\}$，　$T^1 = \{0, 1\}$，

$T^2 = \{00, 01, 10, 11\}$，　…

$T^* = \{\varepsilon, 0, 1, 00, 01, 10, 11, \ldots\}$

$T^+ = \{0, 1, 00, 01, 10, 11, \ldots\}$

语 言 （LANGUAGES）

✧ 概念 设T为字母表，则任何集合$L \subseteq T^*$是字母表T上的一个语言（language）

✧ 举例

- 英文单词集 {…, English, …, words , … }
- C语言程序集{ … } 字母表?
- 汉语成语集{ …, 马到成功, … }
- 化学分子式集{ …, H_2O, …, NaCl , … }
- { any, 任意 }

语 言 （LANGUAGES）

✧举例： 设T = {a，b}

则 $L_1 = \{a^n b^n \mid n \geq 1\}$

$L_3 = \{ b^k \mid k$ 是质数$\}$

$L_2 = \{\varepsilon\}$ 只有一个空句子的语言

$L_4 = \{\} = \Phi$ 空语言

均为字母表T上的语言。

✧由语言的定义知语言是集合，对于集合的运算可应用于对于语言的计算。如并，交，补，差。

语言的基本运算

✧ **语言的积：**

两个语言L_1和L_2的积L_1L_2是由L_1和L_2中的字符串连接所构成的字符串的集合。即L_1中所有字符串分别与L_2中的字符串连接得到的集合。

设T={a, b}，　L_1和 L_2是T上的语言。

L_1 ={ab, ba}　L_2 ={aa, bb}

则 L_1L_2 ={abaa, abbb, baaa, babb}

L_2L_1 ={aaab, aaba, bbab, bbba}

$L_1L_2 \neq L_2L_1$　语言的积不可交换。

语言的基本运算

✧ **语言的幂：**

语言的幂可归纳定义如下:

$L^0 = \{\varepsilon\}$

$L^n = L \cdot L^{n-1} = L^{n-1} \cdot L \qquad n \geq 1$

上例中，

$L_1^2 = \{abab, abba, baab, baba\}$

$L_2^2 = \{aaaa, aabb, bbaa, bbbb\}$

4. 文法

- 启发
- 文法定义
- 短语结构文法：乔姆斯基体系
- 标准形式文法

句子结构的表示

- 几个自然语言的句子：
  - 集合是数学的基础。
  - 中国加入WTO。
  - 形式语言是很抽象的。
- 结构：<名词短语><动词短语><句号>
  - <名词短语>={集合，中国，形式语言}
  - <动词短语>={是数学的基础，加入WTO，是很抽象的}
  - <句号>={。}

结构的嵌套

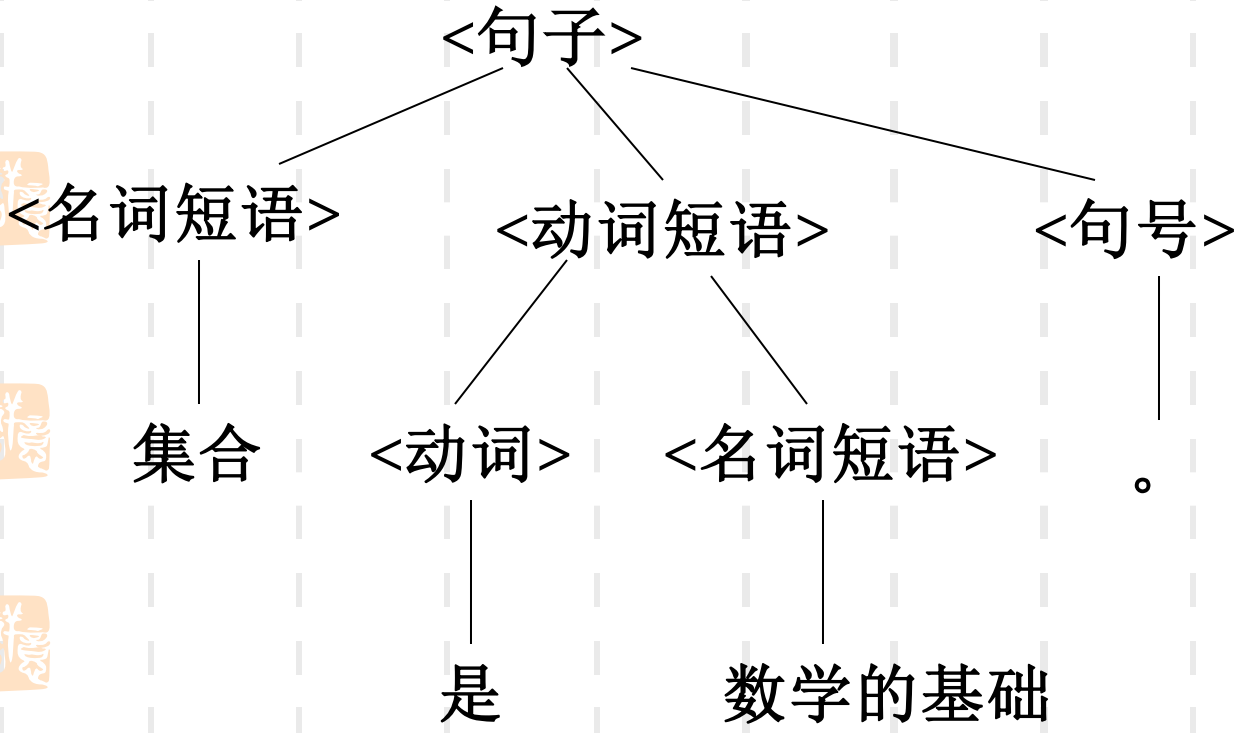
- <动词短语>的结构：
  - <动词><名词短语> (例：加入WTO)
  - <动词><形容词短语> (例：是很抽象的)
- 那么各集合就变为
  - <名词短语>={集合，中国，形式语言，数学的基础， WTO}
  - <动词>={是，加入}
  - <形容词短语>={很抽象的}

完整的文法表示

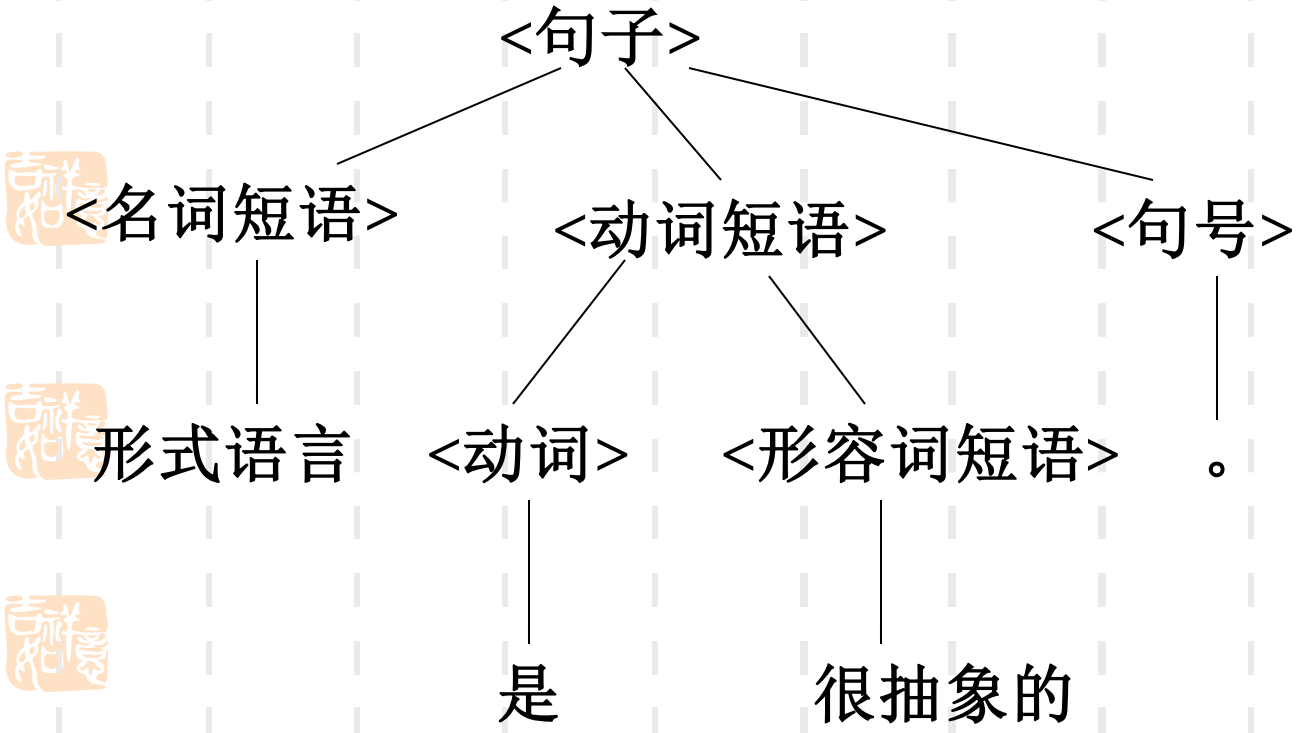
- <句子>→ <名词短语><动词短语><句号>
- <动词短语>→<动词><名词短语>
- <动词短语>→<动词><形容词短语>
- <动词> → 是
- <动词> → 加入
- <形容词短语> →很抽象的
- <名词短语> →集合
- <名词短语> →中国
- <名词短语> →形式语言
- <名词短语> →数学的基础
- <名词短语> →WTO
- <句号>→。

句子图解

- “集合是数学的基础。”图解。



- “形式语言是很抽象的。”图解。



启发

- 表示语言四要素

1. 一系列“符号”，即语法变量，如<名词短语>
2. 最终定义的结构，如<句子>
3. 终极符号，如“中国”
4. 规则，即产生式，如<动词短语> → <动词> <名词短语>

4.2 形式定义

- 定义4-1 文法(grammar)*G*是一个四元组：

$$G = (V, T, P, S)$$

✓ *V*—变量(variable)的非空有穷集。一个语法变量表示了一个语法范畴。

✓ *T*—终极符(terminal)的非空有穷集。$V \cap T = \varnothing$

✓ *P*—产生式(production)的非空有穷集。*P*中元素具有形式$\alpha \to \beta$, 其中$\alpha \in (V \cup T)^+$,且α中至少有*V*中的一个元素出现。 $\beta \in (V \cup T)^*$。 α, β依次称为产生式$\alpha \to \beta$的左部和右部。

✓ *S*—$S \in V$，文法*G*的开始符号(start symbol)

例4-1

- (1) $(\{A\}, \{0, 1\}, \{A \to 01, A \to 0A1, A \to 1A0\}, A)$

合法的句子：{01, 0011, 1010, 000111, …}

- (2) $(\{A\}, \{0, 1\}, \{A \to 0, A \to 0A\}, A)$

合法的句子：{0, 00, 000, …}

- (3) $(\{A, B\}, \{0, 1\}, \{A \to 01, A \to 0A1, A \to 1A0, B \to BA, B \to 0\}, A)$

产生句子的过程中，B不起任何作用。

讨论

例4-1(6)

$(\{S\}, \{a, b\}, \{S \to 00S, S \to 11S, S \to 00, S \to 11\}, S)$

是文法吗？为什么？

不是。产生式右部0, 1等符号既不为语法变量也不为终极符。即不满足定义中$\beta \in (V \cup T)^*$。更改：可将a, b依次改为0, 1 。

约定

- 对一组有相同左部的产生式

$\alpha \to \beta_1, \alpha \to \beta_2, \ldots, \alpha \to \beta_n,$

可以简记为

$\alpha \to \beta_1 | \beta_2 | \ldots | \beta_n$

- $A, B, C, \ldots$表示语法变量
- $a, b, c, \ldots$表示终极符号
- $X, Y, Z, \ldots$表示语法变量或终极符号
- $x, y, z, \ldots$表示终极符号组成的行
- $\alpha, \beta, \gamma, \ldots$表示语法变量或终极符号组成的行

4.3. 短语结构文法

1. 0型文法（无限制）

设文法$G = (V_N, V_T, P, S)$

V_N：非终止符，用大写字母表示

V_T：终止符，用小写字符表示

P：产生式

S：起始符

产生式P：$\alpha \rightarrow \beta$，　其中$\alpha \in V^+$，$\beta \in V^*$　　α，β无任何限制(V^+不包括空格,V^*包括空格)

例：0型文法 $G = (V_N, V_T, P, S)$

$V_N = \{S, A, B\}$

$V_P = \{a, b, c\}$

P: ① $S \rightarrow aAbc$ ②$Ab \rightarrow bA$ ③ $Ac \rightarrow Bbcc$

④ $bB \rightarrow Bb$ ⑤ $aB \rightarrow aaA$ ⑥ $aB \rightarrow \lambda$(空格)

$$S \xrightarrow{①} Aabc \xrightarrow{②} abAc \xrightarrow{③} abBbcc \xrightarrow{④} aBbbcc \xrightarrow{⑥} bbcc$$

此文法可以产生：$X=a^n b^{n+2} c^{n+2}$ $n \geq 0$

$X|_{n=0}=bbcc$

由0型文法产生的语言称为0型语言。

2. 1型文法（上下文有关文法）

设文法$G = (V_N, V_T, P, S)$

产生式P：$\alpha_1 A \alpha_2 \rightarrow \alpha_1 \beta \alpha_2$

其中$A \in V_N$，$\beta \in V^+$, $\alpha_1, \alpha_2 \in V^*$

$|\alpha_1 A \alpha_2| \leq |\alpha_1 \beta \alpha_2|$，或$|A| \leq |B|$

由上下文有关文法构成的语言称为上下文有关语言，用$L(G_1)$表示，G_1：上下文有关文法

例：$G = (V_N, V_T, P, S)$

$V_N = \{S, B, C\}$　$V_T = \{a, b, c\}$

P: ① $S \rightarrow aSBC$　② $CB \rightarrow BC$　③ $S \rightarrow abC$　④ $bB \rightarrow bb$

⑤ $bC \rightarrow bc$　⑥ $cC \rightarrow cc$

$\lambda_1 S \lambda_2 \rightarrow \lambda_1 aSBC \lambda_2$, $bB\lambda \rightarrow bb\lambda$

对于$S \rightarrow aSBC$

∵ $\alpha_1 = \lambda$, $\alpha_2 = \lambda$, $A = S$, $B = aSBC$,并且$|S| < |aSBC|$

∴ 符合1型文法规则

对于$bB \rightarrow bb$

∵ $\alpha_1 = b$, $\alpha_2 = \lambda$, $A = B$, $B = b$,并且$|B| \leq |b|$

∴ 也符合1型文法规则

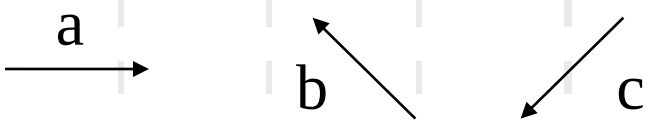
产生式都符合1型文法的要求

$$S \xrightarrow{①} aSBC \xrightarrow{③} aabCBC \xrightarrow{②} abbBCC \xrightarrow{④} aabbCC \xrightarrow{⑤} aabbcC \xrightarrow{⑥} aabbcc$$

$\therefore X=a^2b^2c^2$

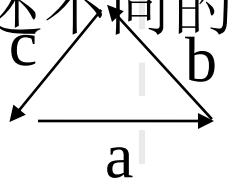
此文法G可产生的语言：$L(G)=\{a^nb^nc^n|n=1,2...\}$

假设基元

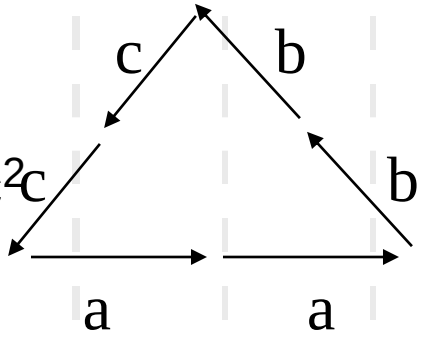


语言L(G)可以描述不同的三角型

$X= abc$



$X= a^2b^2c^2c$



2．2型文法（上下文无关文法）

设文法$G = (V_N, V_T, P, S)$

产生式P：$A \rightarrow \beta$　其中$A \in V_N$（且是单个的非终止符)

$\beta \in V^+$ (可以是终止符，非终止符，不能是空格)

对产生式的限制比较严格

由上下文无关文法构成的语言称为上下文无关语言。

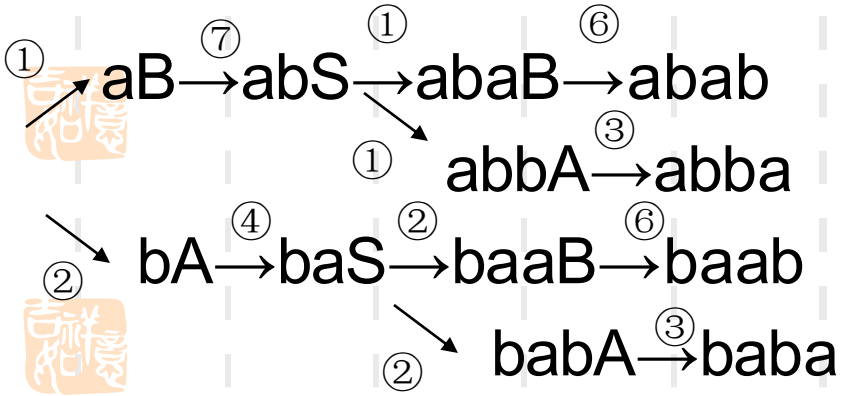
例：文法G = (V_N,V_T, P, S)

V_N = {S, B, C}

V_T = {a, b}

P: ① S→aB ② S→bA ③ A→a ④ A→aS

⑤ A→bAA ⑥ B→b ⑦ B→bS ⑧B→aBB



例：$G = (V_N, V_T, P, S)$

$V_N = \{S, T, F\}$

$V_T = \{a, +, *, (,)\}$

P: ① $S \rightarrow S+T$　② $S \rightarrow T$　③ $T \rightarrow T*F$　④ $T \rightarrow F$

⑤ $F \rightarrow (S)$　⑥ $F \rightarrow a$

$$S \xrightarrow{①} S+T \xrightarrow{②} T+T \xrightarrow{④} F+T \xrightarrow{⑥} a+T \xrightarrow{③} a+T*F \xrightarrow{④} a+F*F \xrightarrow{⑥} a+a*F \xrightarrow{⑥} a+a*a$$

两种方法替换非终止符：

① 最左推导：每次替换都是先从最左边的非终止符开始，例如上边的例子。我们经常采用最左推导。

② 最右推导：每次替换都是先从最右边的非终止符开始，

例如： $S \rightarrow S+T \rightarrow S+F \rightarrow S+a \rightarrow T+a \rightarrow F+a \rightarrow a+a$

3. 3型文法（有限状态文法）

文法 $G = (V_N, V_T, P, S)$

产生式P：$A \rightarrow aB$ 或$A \rightarrow a$， （对产生式限制最严格）

其中$A,B \in V_N$（且是单个字符），$a \in V_T$(且是单个字符)

由3型文法产生的语言成为3型语言。

例：文法$G = (\{S, A\},\{0, 1\}, P, S)$

P: ① $S \rightarrow 0A$ ② $A \rightarrow 0A$ ③ $A \rightarrow 1$

$S \rightarrow 0A \rightarrow 00A \rightarrow 000A \rightarrow 0001$

$L(G)=\{0^n1 | n=1,2...\}$

例：构造文法G能产生语言$L(G) = \{x | x=0^n\ 10^m \mid_{n,m>0}\}$

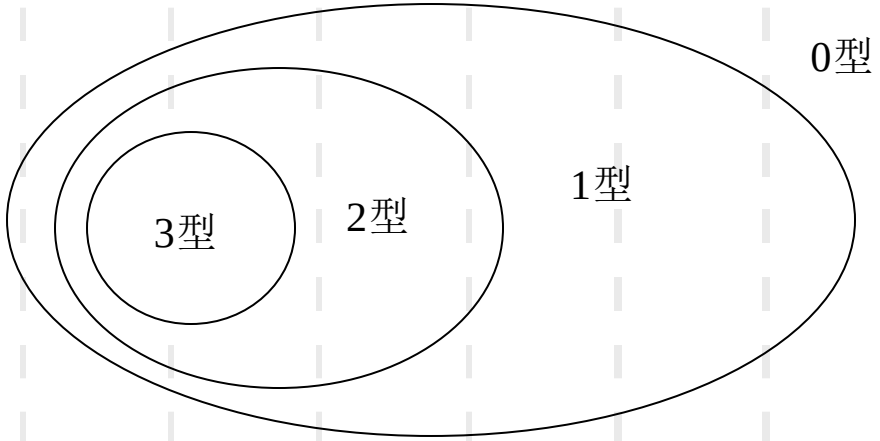
解：$G = (V_N, V_T, P, S)$

$V_T=(0,1)$

P: ① $S \rightarrow 0S$ ② $B \rightarrow 0B$ ③ $S \rightarrow 1B$ ④ $B \rightarrow 0$

$\therefore V_N=(S, B)$

四种文法的关系：



包含关系：限制不严格的文法必然包含限制严格的文法。

分析

- 该体系指明了形式语言研究的方式，0型文法最为一般，3型文法最为特殊。
- 1型文法规定推导过程中句型长度递增；2型文法规定产生式的左边只能为一个语法变量，这样某个语法变量的推导不会依赖于其上下文，这也是“上下文无关语言”，以及与之相对的“上下文有关语言”名称的来历；3型文法的则只允许从一个语法变量最多产生出一个语法变量，且该语法变量只能在句型尾。

文法分类

- 设文法G=(V,T,P,S)，则判断G是哪类文法的方法如下：
  - 1、G是短语结构文法；
  - 2、如果所有产生式都有右边部分长度大于等于左边部分，那么G是上下文有关文法；
  - 3、如果如果所有产生式的左边部分都是单个非终极符号，那么G是上下文无关文法；
  - 4、如果所有产生式的右边部分都是以终极符号开始、含有至多一个非终极符号、如果有非终极符号则出现在最右边，那么G是正规文法。

例 文法的分类

(1) G_1: $S \to 0|\ 1|\ 00|\ 11$

G_3: $S \to 0|\ 1|\ 0A|\ 1B,\ A \to 0,\ B \to 0$

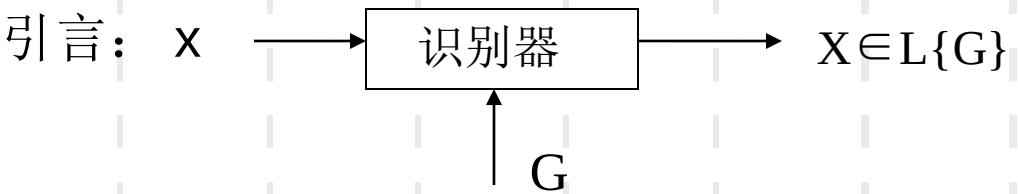
(2) G_5: $S \to A|\ B|\ BB,\ A \to 0,\ B \to 0$

(3) G_{14}: $S \to aBC|\ aSBC$

$aB \to ab,\ bB \to bb,\ bC \to bc,\ cC \to cc$

(4) G_7: $S \to \varepsilon|\ 0S$

§5.3 自动机理论



当给出某类文法后，可根据它设计一种相应的称为自动机的硬件模型。它由控制装置、输入带和某些类型的存储器组成，这种硬件模型是一种识别器称自动机。不同的自动机可以识别不同的文法形成的语言。

0型文法：图灵机识别

上下文有关型：线性约束自动机

上下文无关型：下推自动机

有限状态型：有限状态自动机

一、**有限状态自动机** 可以识别由有限状态文法所构成的语言

1、基本定义：五元式M系统　M=(Σ, Q, δ, q_0, F)

其中：　Σ：输入符号的有限集合

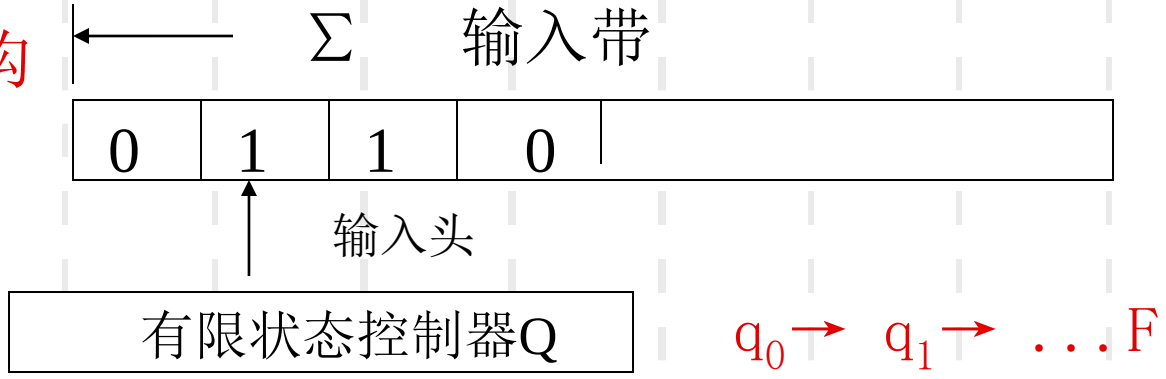
Q：状态的有限集合

δ　：状态转换函数是Q X Σ到Q的幂集一种映射

q_0：初始状态　　$q_0 \in Q$

F：终止状态集合　　$F \subset Q$

2、有限自动机的结构



$q_0 \rightarrow q_1 \rightarrow \ldots F$

转换函数：　δ (q, a)=p

表示有限控制器处于q状态，而输入头读入符号a，则有限控制器转换到下一状态p。

3、自动机识别输入字符串的方式

L(M) = {x| $\delta(q_0, x)$在F中}

$\delta(q, x) = \Phi$　　拒识，停机

4、自动机的状态转换图:表示自动机识别过程

例： $M=(\Sigma, Q, \delta, q_0, F)$

$Q = \{q_0, q_1, q_2, q_3\}$

$\Sigma = \{0, 1\}$

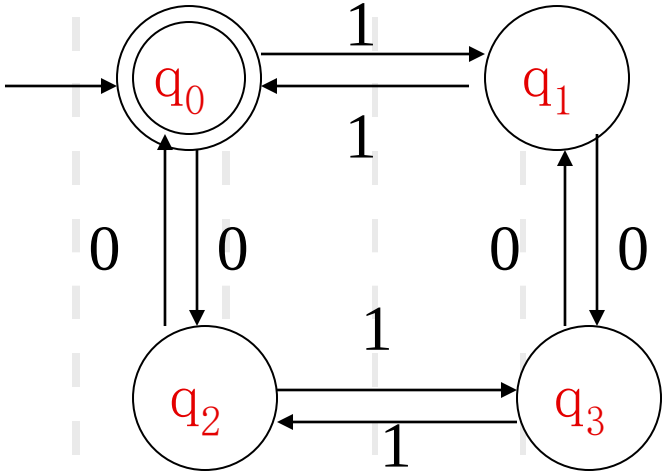
$F = \{q_0\}$

$\delta(q_0, 0) = q_2$, $\delta(q_0, 1) = q_1$,

$\delta(q_1, 0) = q_3$, , $\delta(q_1, 1) = q_0$,

$\delta(q_2, 0) = q_0$, $\delta(q_2, 1) = q_3$,

$\delta(q_3, 0) = q_1$, $\delta(q_3, 1) = q_2$



输入：x=110101

$q_0 \xrightarrow{1} q_1 \xrightarrow{1} q_0 \xrightarrow{0} q_2 \xrightarrow{1} q_3 \xrightarrow{0} q_1 \xrightarrow{1} q_0 \in F$

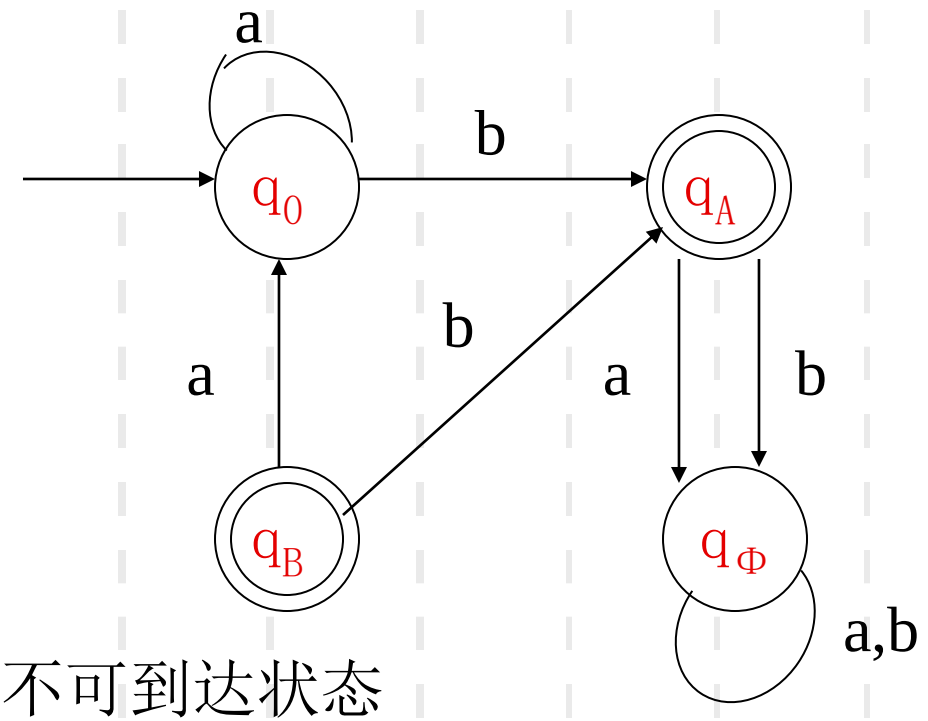
∴ X可以识别

∴ $\delta(q_0, 110101) = q_0$

例：已知自动机状态转换图如下

x_1=aab　可以识别

x_2=abb　不可以识别



可以识别的语言：$L(G)=a^nb$

q_B状态，只有出没有进，为不可到达状态

q_Φ状态，只有进没有出，为陷阱

4、不确定的有限状态自动机

即：$\delta(q, a) = \{q_1, q_2, \ldots q_k\}$ 当输入a时，下一个状态可能为多个状态之一。

例：$M=(\Sigma, Q, \delta, q_0, F)$

$Q = \{q_0, q_1, q_2, q_3, q_4\}$

$\Sigma = \{0, 1\}$

$F = \{q_2, q_4\}$

$\delta(q_0, 0) = \{q_0, q_3\}$，$\delta(q_0, 1) = \{q_0, q_1\}$

$\delta(q_1, 0) = \Phi$（在q_1不会输入0）

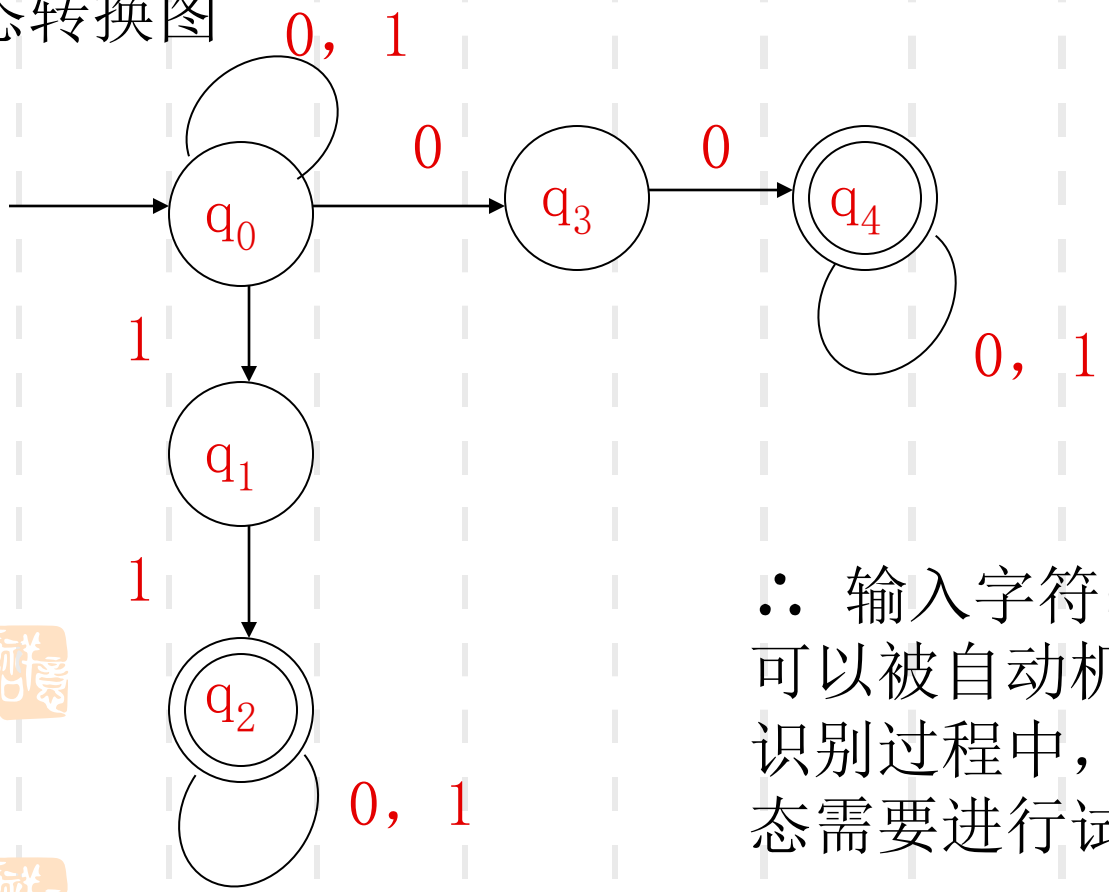
$\delta(q_1, 1) = q_2$

$\delta(q_2, 0) = q_2$，$\delta(q_2, 1) = q_2$，

$\delta(q_3, 0) = q_4$，$\delta(q_3, 1) = \Phi$

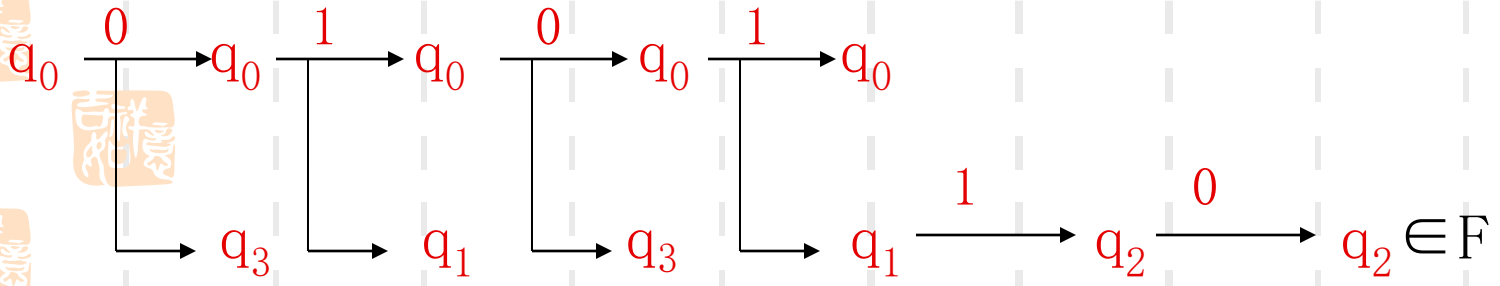
$\delta(q_4, 0) = q_4$，$\delta(q_4, 1) = q_4$

状态转换图



∴ 输入字符串X=010110可以被自动机识别，但在识别过程中，对不确定状态需要进行试探。

输入字串：010110



5、构造一个有限自动机

定理1：设 $G = (V_N, V_T, P, S)$为有限状态文法，一定存在一个有限状态自动机$M=(\Sigma, Q, \delta, S, F)$使$L(G) = L(M)$.

已知有限状态文法$G = (V_N, V_T, P, S)$

由有限状态文法构造有限自动机的步骤：

① $\Sigma = V_T$

② $Q = V_N \cup \{T\}$

③ $q_0 = S$

④ $F = \{T\}$

⑤ 若P中有$B \to a$，则 $\delta(B,a) = \{T\}$, $B \in V_N$, $a \in V_T$

⑥若P中有$B \to aC$，则 $\delta(B,a) = \{C\}$, $B,C \in V_N$, $a \in V_T$

⑦ 对V_T中所有的终止符a，都有 $\delta(T,a) = \Phi$, $a \in V_T$

例：有限状态文法$G = (V_N, V_T, P, S)$　$V_N = \{S, B\}$　$V_T = \{0, 1\}$

P: S→0B, B→0B/ 1S/0　(B→0B , B→1S , B→0)

构造有限自动机 $M=(\Sigma, Q, \delta, q_0, F)$

①∵ $\Sigma =$ V_T　∴ $\Sigma =$ {0, 1}

② ∵ Q = $V_N \cup \{T\} =$ {S, B, T}

③ $q_0 = S$

④ F = {T}

⑤ ∵ S→0B ,∴ δ (S,0) = B,　∵ B→0B ,∴ δ (B,0) = B,

∵ B→1S ,∴ δ (B,1) = S,　∵ B→0 ,∴ δ (B,0) = T,

∵ P中无S→1x ,x∈V_N , ∴ δ (S,1) = Φ

⑥对V_T = {0, 1}有 δ (T,0) = δ (T,1)= Φ

∴构造的自动机M为

$M=(\Sigma, Q, \delta, q_0, F)$，　$\Sigma=\{0,1\}$, $Q=\{S,B,T\}$,

$q_0=\{S\}$, $F=\{T\}$

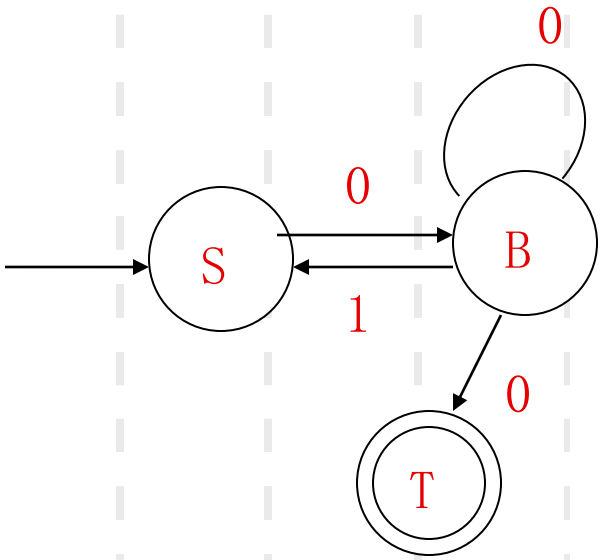
δ：　$\delta(s,0)=\{B\}$

$\delta(s,1)=\Phi$

$\delta(B,0)=\{B,T\}$

$\delta(B,1)=\{S\}$

$\delta(T,0)=\delta(T,1)=\Phi$



设x=00100，可以识别

可以证明：L(M)=L(G)

6.由有限自动机M构造有限状态文法

定理2：已知有限自动机M，则有一个有限状态文法G，使L(G) = L(M)。

已知$M=(\Sigma, Q, \delta, q_0, F)$，构造$G = (V_N, V_T, P, S)$的步骤如下：

① $V_T = \Sigma$　② $V_N = Q$　③ $S = q_0$

④ 对于$\delta(B,a) = C$, 若$B,C \in Q$, $a \in \Sigma$, 则P中有$B \to aC$.

若$C \in F$, 则还有产生式$B \to a$

例：已知有限自动机

$M=(\Sigma, Q, \delta, q_0, F)$，$Q = \{q_0, q_1, q_2\}$

$\Sigma = \{0, 1\}$　　$F = \{q_2\}$

$\delta(q_0, 0) = q_2$, $\delta(q_0, 1) = q_1$, $\delta(q_1, 0) = q_2$,, $\delta(q_1, 1) = q_0$

$\delta(q_2, 0) = q_2$, $\delta(q_2, 1) = q_1$

构造$G = (V_N, V_T, P, S)$ 如下：

①$V_T = \Sigma = \{0, 1\}$

②$V_N = Q = \{q_0, q_1, q_2\}$

③$S = q_0$

④ ∵ $\delta(q_0, 0) = q_2$, ∴有$q_0 \to 0\ q_2$, ∵ $q_2 \in F$ ∴有$q_0 \to 0$

∵ $\delta(q_0, 1) = q_1$, ∴有$q_0 \to 1\ q_1$,

∵ $\delta(q_1, 0) = q_2$, ∴有$q_1 \to 0\ q_2$, ∵ $q_2 \in F$ ∴有$q_1 \to 0$

∵ $\delta(q_1, 1) = q_0$, ∴有$q_1 \to 1 q_0$,

∵ $\delta(q_2, 0) = q_2$, ∴有$q_2 \to 0 q_2$, ∵ $q_2 \in F$ ∴有$q_2 \to 0$

∵ $\delta(q_2, 1) = q_1$, ∴有$q_2 \to 1 q_1$,

∴有限状态文法为：

$G = (V_N, V_T, P, S)$

$V_T = \{0, 1\}$

$V_N = \{q_0, q_1, q_2\}$

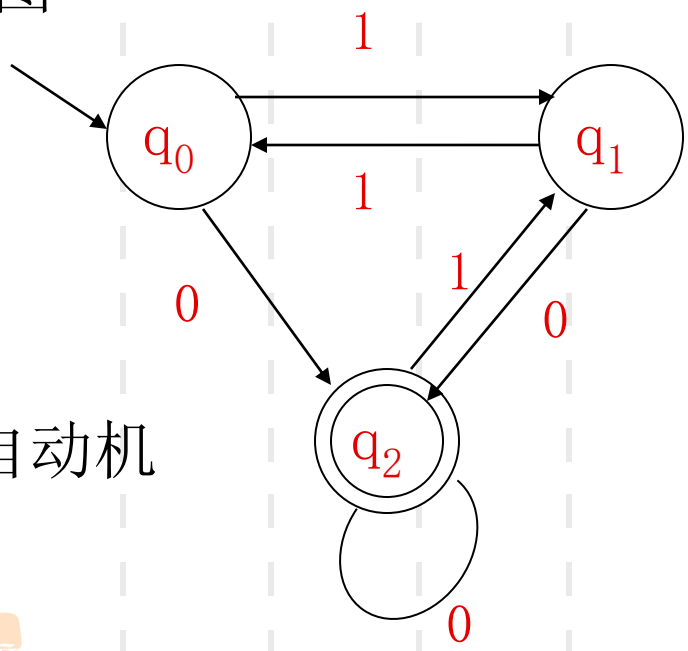
$S = q_0$

P: $q_0 \to 0q_2$, $q_0 \to 1q_1$, $q_0 \to 0$

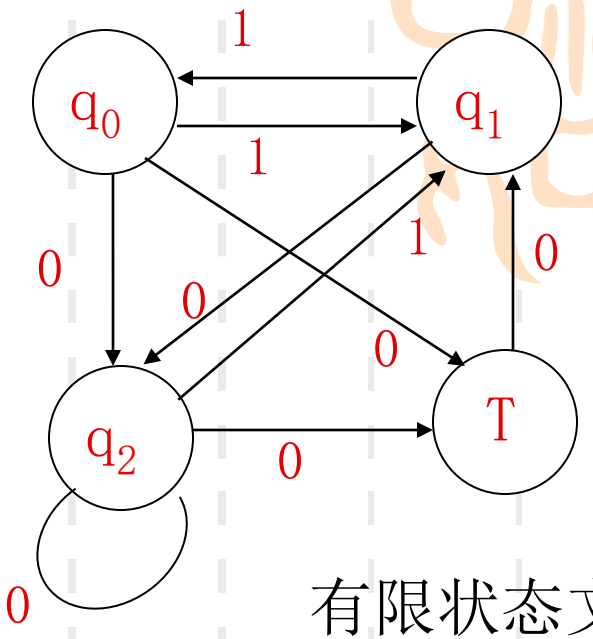
$q_1 \to 0q_2$, $q_1 \to 1q_0$, $q_1 \to 0$

$q_2 \to 0q_2$, $q_2 \to 1q_1$, $q_2 \to 0$

- 状态图



有限自动机



有限状态文法

输入x＝1100

由自动机M： $\delta(q_0,1100)=q_2$， ∵ $q_2 \in F$ ∴1100可以接受

由有限文法G： $q_0 \rightarrow 1q_1 \rightarrow 11q_0 \rightarrow 110q_2 \rightarrow 1100$

∴ L(M) = L(G)

7. NFA与DFA的等价性

- DFA是NFA的特例, 所以NFA必然能接收DFA能接收的语言. 因此证明等价性只要能够证明一个NFA所能接收的语言必能被另一个DFA所接收。

1.定理: 设一个NFA接受语言L, 那么必然存在一个DFA接受L。

2. 证明:

- **策略:对于任意一个NFA,构造一个接收它所能接收语言的DFA，这个DFA的状态对应了NFA的状态集合。**

§5.4 基元的提取

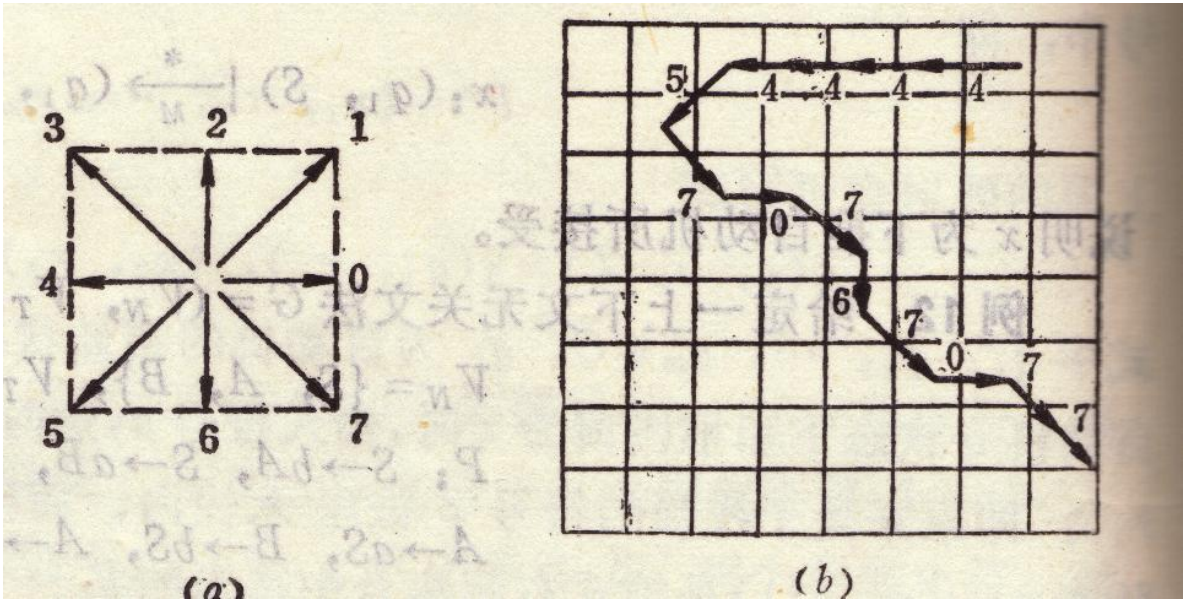
- 模式识别的主要任务之一就是图形的识别。描述一个待识别的图形模式可以借助于一组基元，用语言结构法进行识别。
- 基元对应于文法中的终止符，它是构成句子的最基本单位。
- 选择好的基元对有效地进行句法模式识别是十分重要的，可惜的是基元选择还没有一个通用的方法，只能根据具体因素而定。
  - 模式的数据性质
  - 待识别对象的具体应用
  - 识别系统所用的技术

- 基元选择应注意的两点
  - 基元应是模式的基本单元，且易于利用它们之间的结构关系来紧凑方便地描述模式。
    - 例如：图形识别中各子图形之间的连接关系就可以用基元的结构关系来表达
  - 基元是最简单的子模式，它本身的结构信息已不重要，可用非语言方法（如统计方法、几何尺寸度量等）来提取。
    - 例如：识别手写文字用笔画作为基元比较有效，语音识别采用音素作为基元比较方便，心电图识别采用收缩波和扩张波的波形变化特征作为基元比较直接，等等。

- 在图形识别中，对平面图形可以采用两种类型的基元。
  - 从图形的边界、轮廓或骨架提取基元；
  - 以图形的基本组成区域作为基元。

5.4.1 图形边界或骨架的基元选择

- Freeman链码可以用来描述图形的边界和骨架。



5.4.1 图形边界或骨架的基元选择

- Freeman链码可以用来描述图形的边界和骨架。
  - 将待描述的曲线量化，曲线落在每一个方格中的各个分段可用8个方向之一来近似。
    - 例如，图中的曲线可表示为字符串：4444570767077
  - 可以通过句法分析来判别待识别的曲线。

5.4.1 图形边界或骨架的基元选择

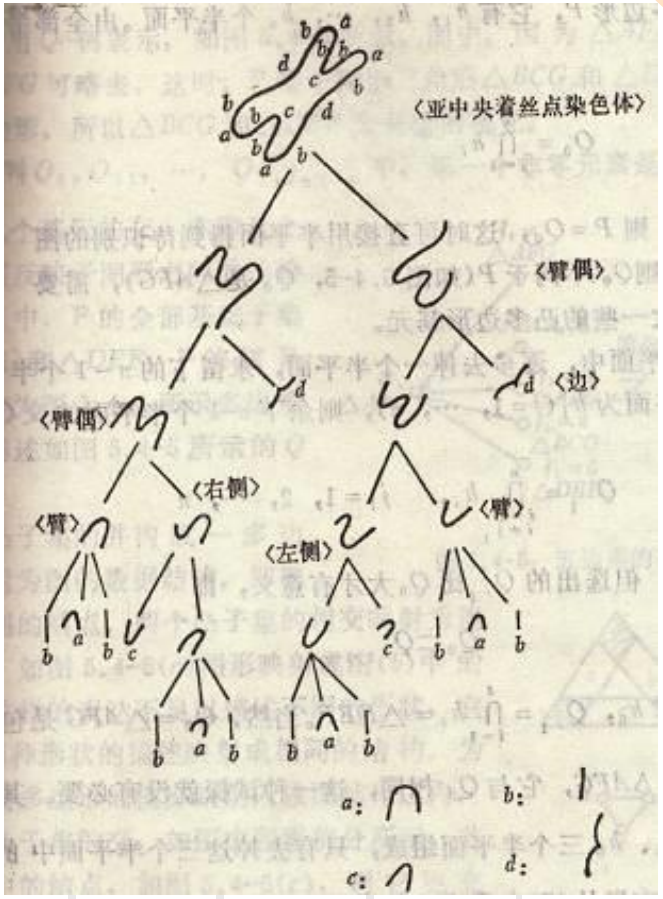
- Freeman链码可以用来描述图形的边界和骨架。
  - 在实际应用中，直接采用Freeman链码中的元素作为基元，会将曲线图形分割的太细，造成句法分析的复杂化。
  - 若选用若干种有共性特点的弧形曲线段作为基元，亦能描述图形，但比直接用Freeman链码简单。
    - 可用一种变换把原来的8种基本方向变换成一组弧形曲线基元，使描述模式的句子短一些。

5.4.1 图形边界或骨架的基元选择

- 可将模式的描述用字符串的形式表示为：$V=V_1 V_2 \ldots V_n$，其中V_i是串中的一个基元，它可以是一段不变曲率的弧，多边形的一条边，或一段二次曲线的弧等等。
- 若V_i取自有穷字母表，则可直接采用句法分析。
  - 进行句法分析之前要消除噪声，对边界曲线进行预处理。

5.4.1 图形边界或骨架的基元选择

- 例子：亚中央着丝点染色体的多级结构描述



5.4.1 图形边界或骨架的基元选择

- 例子：亚中央着丝点染色体的多级结构描述
  - 基元采用曲线段a,b,c,d
  - 从左到右把树的叶子汇集起来，就构成了一个字符串，恰好表达了染色体的边界形状。
  - 用符号编码表示为：babcbabdbabcbabd，表达了这类染色体的一个句子。

5.4.1 图形边界或骨架的基元选择

- 讨论
  - 描述一种图形采用哪种基元最合适没有统一的标准和方法，因图形曲线的不同而异。
  - 一般可兼顾两点
    - 基元的提取方法尽可能简单
    - 描述曲线的句子尽量短一些

5.4.2 按区域划分成多边形近似的基元

- 这种基元着眼于待识别图形的区域。
- 作法
  - 将待识别的图形理想化为一个多边形，再以若干个凸多边形的“并”来表示该多边形，而凸多边形又用若干半平面的“交”来组成。
  - 用形式语言表示
    - 半平面是字母
    - 凸多边形是由这些字母组成的字
    - 由这些字组成的句子就相当于代表图形的多边形
    - 可采用凸多边形作为基元