API - esercitazione 7

13/04/2021

Modelli operazionali: Automi

Modelli generativi: Grammatiche

Modello descrittivo: Logica $\rightarrow$ non è ambigua

FOL

monadiche (MFO / MSO)

FOL

Cosa significa definire un predicato P di $\overbrace{\text{arità}}^{\text{quanti argomenti ha } P}$ n?

$$\forall x_1,\dots,x_n \left(P(x_1,\dots,x_n) \iff \mathcal{F} \right)$$

spiegare quando vale P

Caratteristiche di $\mathcal{F}$:

- formula chiusa
- contiene predicati e funzioni di def. già nota
- Può contenere P ma con argomenti "più semplici" (definizione ricorsiva)

$$\forall x_1, \dots, x_n \quad \big(P(x_1, \dots, x_n) \Leftrightarrow P(x_1, \dots, x_n) \big)$$ Tautologia

Se P ha arità 1, può rappresentare l'appartenenza a un insieme

$$P(x) \equiv x \in P$$

I linguaggi sono predicati di arità 1

$$\forall x \quad \big(\overset{x \in L}{\overset{\equiv}{L(x)}} \Leftrightarrow ♫ \big)$$

es.:

$$L_a = a^*$$

Cosa si può usare:

$=$

$=(x_1, x_2) \equiv x_1 = x_2$

$\cdot$

a, ε

Ricorsione

$$\forall x \; (\overset{x \in L_a}{L_a(x)} \iff x = \varepsilon \vee \exists y \, (y \in L_a \wedge x = a.y))$$

es.: $L = \{a^n b^n c^n \mid n \geq 0\}$

$x = y \cdot z \cdot w$

Cosa si può usare:

▷ $=$

▷ $\cdot$

▷ a, b, c, ε

$L_1 = \{a^n b^n \mid n \geq 0\}$

$L_2 = \{b^n c^n \mid n \geq 0\}$

$L_1 = \{ a^n b^n \mid n \geq 0 \}$ $\qquad$ $L_2 = \{ b^n c^n \mid n \geq 0 \}$

$$\forall x \left(x \in L_1 \iff x = \varepsilon \vee \exists y \left(y \in L_1 \wedge x = a.y.b \right) \right)$$

Idea per la ricorsione: $y \in L_1$ $\qquad$ $y = a^m b^m$

$$x = a.y.b \implies x \in L_1$$

$$S \to aSb \mid \varepsilon$$

$$L_1 = \{ a^n b^n \mid n \geq 0 \} \qquad L_2 = \{ b^n c^n \mid n \geq 0 \}$$

$$\forall x \, (x \in L_1 \iff x = \varepsilon \lor \exists y \, (y \in L_1 \land x = a.y.b))$$

$$\forall x \, (x \in L_2 \iff x = \varepsilon \lor \exists y \, (y \in L_2 \land x = b.y.c))$$

$$L = \{ \underbrace{a^n}_{y} \underbrace{b^n}_{z} \underbrace{c^n}_{w} \mid n \geq 0 \}$$

$$x = y \quad z \quad w$$

$$\forall x \, (x \in L \iff x = \varepsilon \lor \exists y, z, w \, (x = y.z.w \land$$
$$y.z \in L_1 \land z.w \in L_2))$$

$y = aaab$, $z = bb$, $w = cc$

$y.z = aaab.bb \in L_1$

$z.w = bb.cc \in L_2$

$$\forall x \left(x \in L \Leftrightarrow x = \varepsilon \vee \exists y, z, w \left(\overbrace{x = y \cdot z \cdot w} \wedge y \cdot z \in L_1 \wedge z \cdot w \in L_2 \wedge y \in L_a \wedge z \in L_b \wedge w \in L_c \right) \right)$$

$$\forall x \left(\overset{x \in L_a}{L_a(x)} \Longleftrightarrow x = \varepsilon \vee \exists y \left(y \in L_a \wedge x = a \cdot y \right) \right)$$

$$L_3 = \{ a^n c^n \mid n \geq 0 \}$$

$$\forall x \left(x \in L \Leftrightarrow x = \varepsilon \vee \exists y, z, w \left(x = y \cdot z \cdot w \wedge y \cdot z \in L_1 \wedge z \cdot w \in L_2 \wedge y \cdot w \in L_3 \right) \right)$$

vedere a casa

es.:

① Specificare, utilizzando solo il pred. di uguaglianza e la fun. di concatenazione, substring(x, y) che indica che x è sottostringa di y

x	y	substring
ab	abcd	✓
ac	abcd	✗
ε	---	✓

$$\forall x, y \left(\text{substring}(x, y) \iff \exists p, r \left(y = p \cdot x \cdot r \right) \right) \quad \vdots$$

② Specificare substring2(x, y, i, j) indico che x sottostringa di y che va dal carattere in pos. i fino al carattere in pos. j (estremi inclusi)

x	y	i	j	sub2
ab	abcd	Ø	1	✓
ab	abcd	1	2	✗
ε	*	*	*	✗
c	abcd	2	2	✓
*	*	2	1	✗

Cosa utilizzare: $\cdot$, =, op. aritmetiche (+, $\cdot$, /, −) e la fun. unaria len(x)

$$\forall_x \forall_y \forall_i \forall_j \left(\text{substring2}(x, y, i, j) \iff \exists p, r \left(y = p.x.r \;\wedge\; \text{len}(p) = i \;\wedge\; \text{len}(x) = j - i + 1 \right)\right)$$

$$\begin{array}{cccccc} & & 0 & 1 & 2 & 3 \\ y & = & a & b & c & d \\ & & p = & & & = r \end{array}$$

$$\text{len}(y) = 4$$

$$j = \overset{i}{\text{len}(p)} + \text{len}(x) \quad ✗$$

$$x = bc$$

$$i = 1$$

$$j = 2$$

$$1 \qquad 4 \qquad \underset{2}{\text{len}(x)} = j - i + 1 \quad ✓$$

$$\text{len}(r) = \text{len}(y) - j - 1$$

$$j = \text{len}(p) + \text{len}(x) = i + \text{len}(x)$$ ✗

$$\text{len}(x) = j - i + 1$$ ✓

③ reverse(x, y) x è l'inversa di y

Cosa utilizzare: $\cdot$, =, op. aritmetiche (+, $\cdot$, /, −) e

la fun. unaria len(x)

abc cba ✓

ww^R

x					y			
a	b	b	c		c	b	b	a
0	1	2	3		0	1	2	3

$len(y) - i - 1$

$substring2(c, y, i, i)$

$\downarrow$

riusciamo a identificare il carattere in pos. i

$\forall x, y \, (\exists y \, (P(x,y)))$ ✗

$\exists y (P(y)) \wedge \forall y (P'(y))$ ✓

$$\forall x, y \, (reverse(x, y) \iff \forall c \, \forall i \, (substring2(c, x, i, i)$$

$$\implies substring2(c, y, len(y) - i - 1, len(y) - i - 1)) \wedge$$

$$\forall c \, \forall i \, (substring2(c, y, i, i) \implies sub2(c, x, len(x) - i - 1, len(x) - i - 1))$$

$$\forall x, y \Big(reverse(x, y) \Leftrightarrow \forall c \, \forall i \Big(substring2(c, x, i, i) \Rightarrow substring2(c, y, len(y) - i - 1, len(y) - i - 1) \Big) \land len(x) = len(y) \Big)$$

Via ricorsiva :

ww^R : $S \to aSa \mid bSb \mid \varepsilon$

$$\forall x, y \left(reverse(x, y) \Leftrightarrow len(x) = len(y) = 0 \ \vee \right.$$
$$\left. \exists z, w, c \left(x = c.w \ \wedge \ y = z.c \ \wedge \ reverse(w, z) \ \wedge \ len(c) = 1 \right) \right)$$

sulle slide c'è un'alternativa

es.: Pre / Post condizioni di un programma

$$A(\overbrace{n, m}^{IN}) = \overbrace{z}^{OUT} \qquad z = \begin{cases} 1 & n, m \text{ coprimi} \\ 0 & \text{altrimenti} \end{cases}$$

n, m coprimi se $MCD(n, m) = 1$

$K : \frac{n}{k} \in \mathbb{N}, \ \frac{m}{k} \in \mathbb{N}$ $\qquad \frac{6}{3} = 2 \qquad \frac{6}{5} \notin \mathbb{N}$

3 è div. di 6

5 no

PRE: $n, m \in \mathbb{N}$

$$\forall x, y \left(\text{coprimi}(x, y) \iff \neg \exists k, x_1, x_2 \in \mathbb{N} \left(x = k x_1 \land y = k x_2 \land \underbrace{k > 1}_{} \right) \right)$$

POST: $\text{coprimi}(n, m) \implies z = 1$

A	B	A ⇒ B
1	1	1
1	0	0
0	1	1
0	0	1

coprimi	z = 1	A ⇒ B
1	1	1
1	0	0
0	1	1
0	0	1

return 1;

$$z = \begin{cases} 1 & \text{se copr.} \\ 0 & \text{altrim.} \end{cases}$$

POST:

$$\left(\text{coprimi}(n, m) \iff z = 1\right) \wedge \left(z \neq 1 \Rightarrow z = 0\right)$$

Alternativa (tipica nel caso di output con dom. finito):

POST: $\left(\text{coprimi}(n, m) \wedge z = 1\right) \vee \left(\neg\,\text{coprimi}(n, m) \wedge z = 0\right)$

$$\left(\mathcal{J}_1 \wedge z = v_1\right) \vee \left(\mathcal{J}_2 \wedge z = v_2\right) \wedge \dots \wedge \left(\mathcal{J}_k \wedge z = v_k\right)$$

Pre / Post cond. con $z \in \mathbb{N}$ sulle slide

Es.: Specifica di sistemi

Problema delle 8 regine

Posizionare 8 regine su una scacchiera 8×8

nessuna regina deve essere sulla stessa:
- riga
- colonna
- diagonale

Cosa utilizzare:

op. su $\mathbb{N}$ e ordine su $\mathbb{N}$

$p(n, x, y) \equiv$ regina n è posizionata in (x, y)

$|\cdot| : \mathbb{Z} \longrightarrow \mathbb{N}$

Dato che le righe, le colonne e le regine sono 8, tutte le var. $\in \{1, 2, 3, 4, 5, 6, 7, 8\}$

Tutte le regine posizionate:

$$\forall n \; \exists x, y \; (p(n, x, y))$$

Ogni regina non è in più pos.:

$$\forall n, x_1, y_1 \left(p(n, x_1, y_1) \Rightarrow \neg \exists x_2, y_2 \left(p(n, x_2, y_2) \wedge (x_1 \neq x_2 \vee y_1 \neq y_2) \right) \right)$$

Al più una regina in riga x:

$$\forall n_1, x, y_1 \left(p(n_1, x, y_1) \Rightarrow \neg \exists n_2, y_2 \left(p(n_2, x, y_2) \land n_1 \neq n_2 \right) \right)$$

Al più una regina in colonna y:

$$\forall n_1, x_1, y \left(p(n_1, x_1, y) \Rightarrow \neg \exists n_2, x_2 \left(p(n_2, x_2, y) \land n_1 \neq n_2 \right) \right)$$

N

NO NE

$y - x$

$2 - 4 = -2 \notin \mathbb{N}$

O

	1	2	3	4	5
1		1,2		1,4	
2			2,3	2,4	
3		3,2		3,4	
4	4,1	4,2			4,5
5					

E

S

(x_1, y_1) e (x_2, y_2) su NE se $x_1 + y_1 = x_2 + y_2$

(x_1, y_1) e (x_2, y_2) su NO se $x_1 + y_2 = x_2 + y_1$

$$\forall n_1, n_2, x_1, x_2, y_1, y_2 \Big(\big(p(n_1, x_1, y_1) \wedge p(n_2, x_2, y_2) \wedge n_1 \neq n_2 \big) \Longrightarrow \neg \big(\overbrace{x_1 + y_2 = x_2 + y_1}^{NO} \big) \wedge \neg \big(\overbrace{x_1 + y_1 = x_2 + y_2}^{NE} \big) \Big)$$

$$J_1 \quad \wedge$$

$$J_2 \quad \wedge$$

$$\vdots$$

Logica Monadica (MFO/MSO)

Solo per linguaggi

Caratteristiche:

▷ Var. sono def. su un sottoinsieme finito di $\mathbb{N}$
$\{0, 1, \dots, n-1\}$, dove n è la lung. della stringa

▷ formula φ può essere:

- $\neg \varphi$
- $\varphi_1 \wedge \varphi_2$
- $\forall x (\varphi)$
- $i(x)$ $\quad \forall i \in I$
  in pos. x c'è la lettera "i"
  $b(3)$
- $x < y$

- $\varphi_1 \vee \varphi_2 = \neg(\neg\varphi_1 \wedge \neg\varphi_2)$
- $\varphi_1 \Rightarrow \varphi_2 = \neg\varphi_1 \vee \varphi_2$
- $\exists x\,(\varphi) = \neg\forall x\,(\neg\varphi)$
- $x = y$
- $x \leq y$
- costante 0
- Pred. di successore $y = S(x) \equiv y = x+1$
- Costanti num. $1, 2, \ldots$
- last(x)
- $y = x + k$
- $y = x - k$

es.: stringhe che iniziano con "a" e hanno almeno 3 "b"

(MFO)

$$a(0) \wedge \exists x_1, x_2, x_3 \left(b(x_1) \wedge b(x_2) \wedge b(x_3) \wedge x_1 \neq x_2 \wedge x_2 \neq x_3 \wedge x_1 \neq x_3 \right)$$

in pos. 0 c'è "a"

$a(x) \equiv$ in pos. x c'è "a"

$b(x) \equiv$ in pos. x c'è "b"

FOL : $\forall x \; (x = \text{"abc"})$

MFO : $\forall x \in \{0, 1, \ldots, n-1\}$

MSO : $\exists P (\ldots)$