Sistemas Baseados em Conhecimento
Aula de Exercícios V

Vinícius Bitencourt Matos

IME-USP

Setembro de 2017

Exercício 11

O código Prolog a seguir define um predicado p.

```
p(X, [X | Y]).
p(X, [Y | Z]) :- p(X, Z).
```

(a) Mostre as árvores de prova e as soluções correspondentes às consultas `p(A, [2, 1, 3])` e `p(2, [1, A, 3])`.

(b) Que operação padrão de lista p representa?

Exercício 9d (aula anterior)

(d) Prove que West é um criminoso utilizando encadeamento para trás.

Base de conhecimento:

1. $\text{American}(x) \wedge \text{Weapon}(y) \wedge \text{Hostile}(z) \wedge \text{Sells}(x, y, z) \rightarrow \text{Criminal}(x)$
2. $\text{Enemy}(\text{nono}, \text{america})$
3. $\text{Missile}(m)$
4. $\text{Owns}(\text{nono}, m)$
5. $\text{Missile}(v) \wedge \text{Owns}(\text{nono}, v) \rightarrow \text{Sells}(\text{west}, v, \text{nono})$
6. $\text{American}(\text{west})$
7. $\text{Missile}(u) \rightarrow \text{Weapon}(u)$
8. $\text{Enemy}(t, \text{america}) \rightarrow \text{Hostile}(t)$

Exercício 12

O procedimento de encadeamento para frente para satisfatibilidade de cláusulas de Horn pode ser estendido para lidar com negação por falha marcando átomos incrementalmente com Y (quando se sabe que estão resolvidos) e N (quando se sabe que são insolúveis), usando o seguinte procedimento:

Para cada átomo q não marcado,

- se há uma regra $q \leftarrow a_1, \ldots, a_n$ na base de conhecimento, em que todo a_i positivo está marcado com Y e todo a_i negativo está marcado com N, então marque q com Y;
- se, para toda regra $q \leftarrow a_1, \ldots, a_n$ na base de conhecimento, algum a_i positivo estiver marcado com N ou algum a_i negativo estiver marcado com Y, então marque q com N.

Note que o primeiro caso trivialmente se aplica para regras em que $n = 0$, e que o segundo caso trivialmente se aplica se não há regras com q como consequente.

Exercício 12a

(na lousa)

(a) Mostre como este procedimento marcaria os átomos da seguinte base de conhecimento:

- $a \leftarrow$
- $b \leftarrow a$
- $c \leftarrow b$
- $d \leftarrow \mathbf{not}(c)$
- $e \leftarrow c, g$
- $f \leftarrow d, e$
- $f \leftarrow \mathbf{not}(b), g$
- $g \leftarrow \mathbf{not}(h), \mathbf{not}(f)$

Exercício 12b

(na lousa)

(b) Dê um exemplo de uma base de conhecimento para a qual este procedimento não marca um átomo com Y nem com N, mas em que esse átomo é intuitivamente Y, de acordo com negação por falha.

Exercício 13

Esta questão trata de propriedades semânticas de cláusulas de Horn proposicionais. Para qualquer conjunto S de sentenças, seja $\mathcal{I}_S$ a interpretação que satisfaz um átomo p se e somente se $S \models p$.

(a) Mostre que, se S é um conjunto de cláusulas de Horn positivas, então $\mathcal{I}_S \models S$.

(b) Dê um exemplo de conjunto S de cláusulas em que $\mathcal{I}_S \not\models S$.

(c) Seja S um conjunto de cláusulas de Horn positivas, e seja α uma cláusula de Horn negativa. Mostre que, se $\mathcal{I}_S \not\models \alpha$, então $S \cup \{\alpha\}$ é insatisfatível.

(d) Sejam S e T conjuntos de cláusulas de Horn positivas e negativas, respectivamente. Usando (c), mostre que, se $S \cup \{\alpha\}$ é satisfatível para todo $\alpha \in T$, então $S \cup T$ também é satisfatível.

(e) No caso proposicional, podemos dizer que o interpretador de Prolog recebe um conjunto de cláusulas positivas S (o programa) e uma única cláusula negativa α (a consulta), e determina se $S \cup \{\alpha\}$ é satisfatível. Use (d) para concluir que o Prolog pode ser usado para testar a satisfatibilidade de um conjunto arbitrário de cláusulas de Horn.

Exercício 13a

(a) Mostre que, se S é um conjunto de cláusulas de Horn positivas, então $\mathcal{I}_S \models S$.

Exercício 13a

Solução

(a) Mostre que, se S é um conjunto de cláusulas de Horn positivas, então $\mathcal{I}_S \models S$.

Queremos mostrar que $\mathcal{I}_S \models S$, isto é, que $\mathcal{I}_S \models \alpha$ para toda sentença $\alpha \in S$.

Seja α uma sentença qualquer de S. Temos que α é da forma $[\neg p_1, \ldots, \neg p_k, q]$ (com $k \geq 0$).

Se $S \models \{p_1, \ldots, p_k\}$, então $S \models q$, pois $\alpha \in S$; assim, $\mathcal{I}_S \models q$, o que implica que $\mathcal{I}_S \models \alpha$.

Caso contrário, então há algum p_i tal que $S \not\models p_i$; consequentemente, $\mathcal{I}_S \models \neg p_i$, o que também implica que $\mathcal{I}_S \models \alpha$.

Portanto, como $\mathcal{I}_S \models \alpha$ para qualquer sentença $\alpha \in S$, então $\mathcal{I}_S \models S$.

Exercício 13b

(b) Dê um exemplo de conjunto S de cláusulas em que $\mathcal{I}_S \not\models S$.

Exercício 13b

Solução

(b) Dê um exemplo de conjunto S de cláusulas em que $\mathcal{I}_S \not\models S$.

$$S = \{\ p \vee q\ \}$$

Exercício 13c

(c) Seja S um conjunto de cláusulas de Horn positivas, e seja α uma cláusula de Horn negativa. Mostre que, se $\mathcal{I}_S \not\models \alpha$, então $S \cup \{\alpha\}$ é insatisfatível.

Exercício 13c

Solução

(c) Seja S um conjunto de cláusulas de Horn positivas, e seja α uma cláusula de Horn negativa. Mostre que, se $\mathcal{I}_S \not\models \alpha$, então $S \cup \{\alpha\}$ é insatisfatível.

A cláusula α é da forma $\neg p_1 \vee \cdots \vee \neg p_k$.

$\mathcal{I}_S \not\models \alpha$	$S \models \neg\neg(p_1 \wedge \cdots \wedge p_k)$
$\mathcal{I}_S \models \{p_1, \ldots, p_k\}$	$S \models \neg(\neg p_1 \vee \cdots \vee \neg p_k)$
$S \models \{p_1, \ldots, p_k\}$	$S \models \neg\alpha$
$S \models p_1 \wedge \cdots \wedge p_k$	Portanto, $S \cup \{\alpha\}$ é insatisfatível.

Exercício 13d

(d) Sejam S e T conjuntos de cláusulas de Horn positivas e negativas, respectivamente. Usando (c), mostre que, se $S \cup \{\alpha\}$ é satisfatível para todo $\alpha \in T$, então $S \cup T$ também é satisfatível.

Exercício 13d

Solução

(d) Sejam S e T conjuntos de cláusulas de Horn positivas e negativas, respectivamente. Usando (c), mostre que, se $S \cup \{\alpha\}$ é satisfatível para todo $\alpha \in T$, então $S \cup T$ também é satisfatível.

Suponhamos que $S \cup \{\alpha\}$ seja satisfatível para todo $\alpha \in T$. Logo, para todo $\alpha \in T$, segue do item (c) que $\mathcal{I}_S \vDash \alpha$. Então, $\mathcal{I}_S \vDash T$. Além disso, do item (a), $\mathcal{I}_S \vDash S$; logo, $\mathcal{I}_S \vDash S \cup T$. Portanto, $S \cup T$ é satisfatível.

Exercício 13e

(e) No caso proposicional, podemos dizer que o interpretador de Prolog recebe um conjunto de cláusulas positivas S (o programa) e uma única cláusula negativa α (a consulta), e determina se $S \cup \{\alpha\}$ é satisfatível. Use (d) para concluir que o Prolog pode ser usado para testar a satisfatibilidade de um conjunto arbitrário de cláusulas de Horn.

Exercício 13e

Solução

(e) No caso proposicional, podemos dizer que o interpretador de Prolog recebe um conjunto de cláusulas positivas S (o programa) e uma única cláusula negativa α (a consulta), e determina se $S \cup \{\alpha\}$ é satisfatível. Use (d) para concluir que o Prolog pode ser usado para testar a satisfatibilidade de um conjunto arbitrário de cláusulas de Horn.

Dado um conjunto qualquer de cláusulas de Horn, podemos separá-lo em $S \cup T$, em que S contém as cláusulas positivas e T as negativas. Provamos no item (d) que, se $S \cup \{\alpha\}$ é satisfatível para todo $\alpha \in T$, então $S \cup T$ também é satisfatível. A recíproca é trivialmente verdadeira: se $S \cup T$ é satisfatível, então para todo $\alpha \in T$ temos que $S \cup \{\alpha\} \subseteq S \cup T$, logo $S \cup \{\alpha\}$ também é satisfatível. Portanto, para verificar a satisfatibilidade de um conjunto qualquer de sentenças, podemos separá-lo como acima, fixar S como o programa, e usar cada cláusula de T como consulta: se $S \cup \{\alpha\}$ for satisfatível para todo $\alpha \in T$, então $S \cup T$ é satisfatível; caso contrário, $S \cup T$ é insatisfatível.

Referências

Exercício 11 adaptado do exercício 9.21 de *Artificial Intelligence – A Modern Approach* (Stuart J. Russell & Peter Norvig, 2010).
Exercícios 12 e 13 adaptados dos exercícios 6.1 (a, b) e 5.4 de *Knowledge Representation and Reasoning* (Brachman & Levesque, 2004).