

Lógica Computacional

Aula Teórica 22: Unificação

António Ravara

Departamento de Informática

16 de Maio de 2011

Revisão do procedimento

Exemplo em Primeira Ordem

Considere-se o seguinte conjunto de cláusulas, assumindo as variáveis universalmente quantificadas.

$$\{\{\neg Q(x, y), P(f(x), y)\}, \{\neg P(f(x), y), R(x, y, z)\}\}$$

- Um resolvente das duas cláusulas em cima é a cláusula $R_1 = \{\neg Q(x, y), R(x, y, z)\}$.
- Considere-se agora a cláusula $\{\neg P(z, y), R(x, y, z)\}$. Não se consegue resolve-la directamente com a primeira cláusula do conjunto acima, mas substituindo $f(x)$ em z obtém-se a cláusula $\{\neg P(f(x), y), R(x, y, f(x))\}$ que já permite encontrar um resolvente: $R_2 = \{\neg Q(x, y), R(x, y, f(x))\}$.
- Note-se que R_2 é consequência de R_1 : se esta é satisfeita (para qualquer z), então é satisfeita para $z = f(x)$.

Cláusulas de Primeira Ordem

Definição

Considere-se uma fórmula $\varphi \in F_{\Sigma}^X$ tal que $\text{FNS}(\varphi)$, i.e.,

$$\varphi = \forall x_1 \dots \forall x_n \psi$$

sendo ψ uma fórmula de primeira ordem sem quantificadores tal que $\text{FNC}(\psi)$.

- Como todas as variáveis estão universalmente quantificadas (as livres estão-o *implicitamente*), φ pode ser escrita como um conjunto de cláusulas.
- Seja $\mathcal{C}(\psi)$ o conjunto de cláusulas que se obtém de ψ (que está em FNC). Define-se $\mathcal{C}(\varphi) = \mathcal{C}(\psi)$.

Cláusulas de Primeira Ordem

Lema

- Para qualquer $\varphi \in F_{\Sigma}^X$ tal que $FNS(\varphi)$, existe um único $\mathcal{C}(\varphi)$
- Para quaisquer $\varphi, \psi \in F_{\Sigma}^X$, se $\mathcal{C}(\varphi) = \mathcal{C}(\psi)$ então $\varphi \equiv \psi$.

Estes resultados derivam dos respectivos da Lógica Proposicional.

Literais

Na Lógica de Primeira Ordem chamam-se literais às fórmulas atómicas (\perp ou predicados) ou à sua negação (\top ou negação de predicados).

Motivação

Substituição

- No exemplo atrás, encontramos uma substituição (z por $f(x)$) que converteu $\{\neg P(z, y), R(x, y, z)\}$ em $\{\neg P(f(x), y), R(x, y, f(x))\}$, permitindo encontrar um resolvente de duas cláusulas.
- Dado um conjunto de literais ocorrendo em duas cláusulas, para encontrar um resolvente é necessário encontrar substituições que façam iguais literais envolvendo o mesmo predicado.

Exemplo

Sejam $\mathcal{L} = \{P(f(x), y), P(z, w)\}$ e $sub = \{f(x)/z\}\{w/y\}$. Então

$$\mathcal{L}sub = \{P(f(x), y), P(f(x), w)\}\{w/y\} = \{P(f(x), w)\}$$

Unificação

Definição

Um conjunto de literais \mathcal{L} é *unificável* se existe uma substituição *sub* que aplicada a \mathcal{L} torna o conjunto singular (i.e., os vários literais convertem-se num só).

Unificações não são necessariamente únicas

Seja $\mathcal{L} = \{P(f(x), y), P(z, w)\}$.

- Vimos que se $sub_1 = \{f(x)/z\}\{w/y\}$ então $\mathcal{L}sub_1 = \{P(f(x), w)\}$.
- Claro que se $sub_2 = \{w/y\}\{f(x)/z\}$ também $\mathcal{L}sub_2 = \{P(f(x), w)\}$.
- Mas se $sub_3 = \{f(x)/z\}\{a/x\}\{b/y\}$ então $\mathcal{L}sub_3 = \{P(f(x), y), P(f(x), b)\}\{a/x\}\{b/y\} = \{P(f(a), y), P(f(a), b)\}\{b/y\} = \{P(f(a), b)\}$.

Unificador mais geral

Definição

Dado um conjunto de literais \mathcal{L} , a substituição sub é o *unificador mais geral* de \mathcal{L} (e escreve-se $umg(\mathcal{L})$), se é um unificador de \mathcal{L} e se qualquer outro unificador sub' de \mathcal{L} é tal que $subsub' = sub'$.

Proposição

Um conjunto finito de literais é unificável se e só se tem um unificador mais geral.

Prova-se a proposição apresentando um algoritmo que, dado um conjunto finito de literais, ou retorna a mensagem “não unificável” ou retorna o seu unificador mais geral.

Algoritmo de unificação

Seja \mathcal{L} um conjunto finito de literais e faça-se $(\mathcal{L}_0, sub_0) = (\mathcal{L}, \emptyset)$.

Para dado $k \geq 0$, se \mathcal{L}_k é singular então existem sub_i para $1 \leq i \leq k$ tal que $sub_0 sub_1 \cdots sub_k$ é o unificador mais geral de \mathcal{L}_k .

Caso contrário, existem $L_i, L_j \in \mathcal{L}$ tal que para $P \in SP_n$

$$L_i = P(a_1, \dots, a_{m-1}, a_m, \dots, a_n) \text{ e}$$

$$L_j = P(a_1, \dots, a_{m-1}, a'_m, \dots, a'_n).$$

Suponha-se que o l -ésimo símbolo de a_m é a variável x e o de a'_m é o termo t (que não contém x). Então,

$$sub_{k+1} = \{t/x\} \text{ e } \mathcal{L}_{k+1} = \mathcal{L}_k sub_{k+1}$$

e itera-se este processo.

Se nenhuma das condições anteriores se verifica, o algoritmo retorna “ \mathcal{L} não é unificável”.

Exemplo de aplicação do algoritmo de unificação

Seja $\mathcal{L} = \{R(f(g(x)), a, x), R(f(g(b)), a, b), R(f(y), z, b)\}$ e $(\mathcal{L}_0, sub_0) = (\mathcal{L}, \emptyset)$.

Fazendo $sub_1 = \{g(b)/y\}$ obtém-se

$$\mathcal{L}_1 = \mathcal{L}_0 sub_1 = \{R(f(g(x)), a, x), R(f(g(b)), a, b), R(f(g(b)), z, b)\}$$

Como \mathcal{L}_1 não é singular, procura-se nova substituição. Fazendo $sub_2 = \{b/x\}$ obtém-se

$$\mathcal{L}_2 = \mathcal{L}_1 sub_2 = \{R(f(g(b)), a, b), R(f(g(b)), z, b)\}$$

Como \mathcal{L}_2 não é singular, procura-se nova substituição. Fazendo $sub_3 = \{a/z\}$ obtém-se

$$\mathcal{L}_3 = \mathcal{L}_2 sub_3 = \{R(f(g(b)), a, b)\}$$

Como \mathcal{L}_3 é singular, o unificador mais geral de \mathcal{L} é $sub = sub_0 sub_1 sub_2 sub_3$.

Outro exemplo de aplicação do algoritmo de unificação

Seja $\mathcal{L} = \{R(f(g(x)), a, x), R(f(g(a)), a, b), R(f(y), a, b)\}$ e $(\mathcal{L}_0, sub_0) = (\mathcal{L}, \emptyset)$.

Fazendo $sub_1 = \{g(a)/y\}$ obtém-se

$$\mathcal{L}_1 = \mathcal{L}_0 sub_1 = \{R(f(g(x)), a, x), R(f(g(a)), a, b)\}$$

Como \mathcal{L}_1 não é singular, procura-se nova substituição.

Fazendo $sub_2 = \{a/x\}$ obtém-se

$$\mathcal{L}_2 = \mathcal{L}_1 sub_2 = \{R(f(g(a)), a, a), R(f(g(a)), a, b)\}$$

Como \mathcal{L}_2 não é singular, procura-se nova substituição.

Como não há mais variáveis, não há nenhuma substituição que “unifique” os literais. Logo, o algoritmo retorna \mathcal{L} “não unificável”.

Outro exemplo de aplicação do algoritmo de unificação

Seja $\mathcal{L} = \{R(f(g(x)), a, b), R(f(g(a)), a, b), R(f(x), a, b)\}$ e $(\mathcal{L}_0, sub_0) = (\mathcal{L}, \emptyset)$.

Fazendo $sub_1 = \{g(a)/x\}$ obtém-se

$$\mathcal{L}_1 = \mathcal{L}_0 sub_1 = \{R(f(g(g(a))), a, b), R(f(g(a)), a, b)\}$$

Como \mathcal{L}_1 não é singular, procura-se nova substituição.

Como não há mais variáveis, não há nenhuma substituição que “unifique” os literais. Logo, o algoritmo retorna \mathcal{L} “não unificável”.

Em geral, se \mathcal{L} contém um literal como $P(x)$ e outro como $P(f(x))$, não será unificável.

Correcção do algoritmo de unificação

Prova

Note-se que como o conjunto de literais \mathcal{L} é finito, o número de variáveis que ocorrem em \mathcal{L} também o é. Logo, o algoritmo termina sempre após um número finito de passos.

Caso o algoritmo retorne \mathcal{L} “não unificável”, pelos casos analisados atrás vê-se que \mathcal{L} o é de facto.

Assuma-se então que o algoritmo retorna como unificador mais geral $sub = sub_0 sub_1 \cdots sub_k$, com $k \geq 0$. Falta mostrar que sub é de facto o unificador mais geral.

Seja sub' outro unificador de \mathcal{L} . Como $sub_0 = \emptyset$ tem-se que $sub_0 sub' = sub'$. Suponha-se que $sub_0 \cdots sub_n sub' = sub'$, para algum $0 \leq n \leq k$; então $\mathcal{L}_n sub' = \mathcal{L} sub_0 \cdots sub_n sub' = \mathcal{L} sub'$, que é singular, ou seja, se sub' unifica \mathcal{L} também unifica \mathcal{L}_n .

Correcção do algoritmo de unificação

Conclusão da prova

A prova termina por indução natural em n : considere-se que

- 1 $sub_{n+1} = \{t/x\}$ (onde x não ocorre em t); e que
- 2 para $L_i, L_j \in \mathcal{L}$ se tem que para $P \in SP_n$
 $L_i = P(a_1, \dots, a_{m-1}, x, \dots, a_n)$ e
 $L_j = P(a_1, \dots, a_{m-1}, t, \dots, a'_n)$.

Como por hipótese sub' é unificador de \mathcal{L}_n , tem-se que $xsub' = tsub'$, logo $sub_{n+1}sub' = \{t/x\}sub' = sub'$.

Por hipótese de indução, para qualquer $n < k$ tem-se que $sub_0 \cdots sub_{n+1}sub' = sub'$, logo também $sub_0 \cdots sub_ksub' = sub'$ e portanto $sub_0 \cdots sub_k$ é o $unf(\mathcal{L})$.