Lógica Computacional (CC2003) Lógica e Programação (CC216)

Nelma Moreira

Lógica Computacional 22

Conteúdo

1 Introdução à Programação em Lógica

Programação em Lógica

Um programa é um conjunto de fórmulas lógicas e que a sua execução corresponde a uma demonstração de que uma fórmula é um teorema.

Os sistemas que vamos ver baseiam-se em:

- considerar fórmulas em forma prenexa e em que a matriz está em normal conjuntiva: cláusulas
- como sistema dedutivo usar variantes da resolução
- o sistema dedutivo é íntegro e completo, para alguma estrutura.
- computacionalmente "universal", i.e., equivalente a máquinas de Turing ou uma qualquer linguagem de programação de uso geral... C, Java, Python, Haskell

1.1 Resolução para a lógica proposicional

Resolução para a lógica proposicional

Um literal é uma fórmula atómica ou a sua negação: p, $\neg p$

Uma cláusula é uma disjunção de literais: $p \vee \neg q \vee \neg p \vee s$ e pode representar-se por um conjunto $\{p, \neg q, \neg p, s\}$

Então uma fórmula da lógica proposicional em FNC, p.e.

$$\neg p \ \wedge \ (q \ \vee \ r \ \vee \ q) \ \wedge \ (\neg r \ \vee \ \neg s) \ \wedge \ (p \ \vee \ s) \ \wedge \ (\neg q \ \vee \ \neg s)$$

pode ser vista como um conjunto de cláusulas:

$$\{\{\neg p\}, \{q, r\}, \{\neg r, \neg s\}, \{p, s\}, \{\neg q, \neg s\}\}$$

Sistema dedutivo por Resolução (LP)

É um sistema dedutivo por refutação: para deduzir ϕ, deduz-se que $\neg\phi$ é uma contradição.

A consequência semântica é preservada: se $\Sigma \models \phi$ sse $\Sigma \cup \{\neg\phi\}$ não é satisfazível.

Seja Σ um conjunto de cláusulas e representamos por $\mathbf{F}$ a cláusula vazia.

O sistema dedutivo por *resolução* não tem axiomas e apenas uma regra de inferência:

Regra de inferência da Resolução

$$\frac{C \cup \{p\} \qquad C' \cup \{\neg p\}}{C \cup C'}$$

A conclusão $C \cup C'$ diz-se a resolvente das premissas.

Uma dedução de $\mathbf{F}$ a partir de um conjunto $\mathcal{C}$ de cláusulas diz-se uma refutação de $\mathcal{C}$.

Uma dedução por resolução de uma fórmula ϕ é uma refutação de cláusulas que correspondem à FNC de $\neg\phi$.

Dado

$$\{\{\neg p\}, \{q, r\}, \{\neg r, \neg s\}, \{p, s\}, \{\neg q, \neg s\}\}$$

temos

$$\frac{\frac{\{p,s\} \ \{\neg p\}}{\{s\}} \quad \frac{\frac{\{q,r\} \ \{\neg r, \neg s\}}{\{q, \neg s\}} \ \{\neg q, \neg s\}}{\{\neg s\}}}{\mathbf{F}}$$

Integridade e Completude da Resolução para LP

Teorema 22.1. *(Integridade) Seja $\mathcal{C} = \{C_1, \ldots, C_n\}$ um conjunto não vazio de cláusulas da lógica proposicional.*

i. Sejam C_i e C_j cláusulas de $\mathcal{C}$ e R uma resolvente de C_i e C_j. Então $\mathcal{C} \models R$.

ii. Se $\mathcal{C} \vdash_R \boldsymbol{F}$,isto é, existe uma dedução de $\boldsymbol{F}$ a partir de $\mathcal{C}$ usando apenas a regra da Resolução, então $\mathcal{C}$ não é satisfazível.

Teorema 22.2. *(Completude) Se um conjunto de cláusulas $\mathcal{C}$ é não satisfazível então existe uma dedução $\mathcal{C} \vdash_R \boldsymbol{F}$.*

Notação clausal

Um cláusula
$$\{\alpha_1, \ldots, \alpha_k, \neg\beta_1, \ldots, \neg\beta_l\}$$
é equivalente a

$$(\beta_1 \ \wedge \ \cdots \ \wedge \ \beta_l) \rightarrow (\alpha_1 \ \vee \ \cdots \ \vee \ \alpha_k)$$

também se pode representar na notação clausal:

$$\alpha_1, \ldots, \alpha_k \leftarrow \beta_1, \ldots, \beta_l$$

1.2 Cláusulas

Cláusulas para LPO

Seja $\mathcal{L}$ uma LPO com igualdade e pelo menos uma constante.
Um literal positivo (ou átomo) é uma fórmula atómica : $P(x, y)$, $f(x) = a$
Um literal negativo é a negação de uma fórmula atómica: $\neg P(x, y)$
Uma cláusula é uma fórmula ϕ da forma:

$$\forall x_1 \ldots \forall x_s(\alpha_1 \ \vee \ \ldots \ \vee \ \alpha_k \ \vee \ \neg\beta_1 \ \vee \ \ldots \ \vee \ \neg\beta_n)$$

onde α_i, β_i são átomos e $x_1, \ldots, x_s$ todas as variáveis que ocorrem em ϕ.
Também se pode escrever ϕ como:

$$\forall x_1 \ldots \forall x_s((\beta_1 \ \wedge \ \ldots \ \wedge \ \beta_n) \rightarrow (\alpha_1 \ \vee \ \ldots \ \vee \ \alpha_k))$$

ou ainda representá-la em notação clausal:

$$\alpha_1, \ldots, \alpha_k \leftarrow \beta_1, \ldots, \beta_n$$

Literais e Cláusulas Fechadas

Um literal diz-se fechado se todos os seus termos são fechados, isto é, não tem variáveis.

$$\neg P(f(a), c),\ Q(g(f(a), f(b)),\ h(a, g(b)) = f(g(a, f(a)),\ \neg a = b\ ,\ a = f(a)$$

Uma cláusula diz-se fechada se todos os seus literais são fechados.

$$\{\neg P(f(a), c), Q(g(f(a), f(b)), \neg h(a, g(b)) = f(g(a, f(a))\}$$

1.3 Conversão para forma clausal

Conversão em forma clausal

Para qualquer fórmula ϕ existe uma conjunto de cláusulas que é satisfazível se e só se ϕ é satisfazível.

Para converter ϕ para forma clausal primeiro converte-se para forma prenexa:

$$Q_1 x_1 Q_2 x_2 \dots Q_n x_n \psi$$

onde cada Q_i é ou $\forall$ ou $\exists$, $1 \leq i \leq n$, e ψ é uma fórmula sem quantificadores. Depois temos de *eliminar* quantificadores existênciais.

Proposição 22.1. *Seja $\forall y_1 \dots y_n \exists x \phi$ uma proposição de uma linguagem de $1^{\underline{a}}$ ordem $\mathcal{L}$ e seja $\mathcal{L}'$ uma linguagem com os símbolos de $\mathcal{L}$ e com mais um símbolo n-ário f. Então, $\forall y_1 \dots y_n \exists x \phi$ é satisfazível em $\mathcal{L}$ se e só se $\forall y_1 \dots y_n \phi[f(y_1, \dots, y_n)/x]$ é satisfazível em $\mathcal{L}'$.*

Nota: as duas fórmulas não são semanticamente equivalentes: apenas a satisfazibilidade de uma é garantida pela satisfazibilidade da outra.

Conversão em forma clausal

Dem.

Temos que mostrar que existe uma estrutura $\mathcal{A}$ tal que $\mathcal{A} \models \forall y_1 \dots y_n \exists x \phi$ sse existe $\mathcal{A}'$ (estrutura de $\mathcal{L}'$) tal que $\mathcal{A}' \models \forall y_1 \dots y_n \phi[f(y_1, \dots, y_n)/x]$

($\Rightarrow$)Para todos os $a_1, \dots a_n \in A$, existe $b \in A$ tal que

$$\mathcal{A} \models_{s[a_1/y_1]\dots[a_n/y_n][b/x]} \phi$$

b depende dos $a_1, \dots, a_n$.Então a estrutura $\mathcal{A}'$ pode ser igual a $\mathcal{A}$ e tal que o valor de $f^{\mathcal{A}'}$ seja dado por, para todo $a_1, \dots, a_n \in A$:

$$f^{\mathcal{A}'}(a_1, \dots, a_n) = b$$

($\Leftarrow$)

Em $\mathcal{A}'$, consideremos os valores de $f^{\mathcal{A}'}(a_1, \dots, a_n)$. Estes são os valores de cada um dos b.

Skolemização

É um algoritmo que converte uma proposição ϕ de uma linguagem de 1ª ordem, numa conjunção (ou conjunto) de cláusulas $\Phi = \phi_1 \wedge \ldots \wedge \phi_n$ (duma linguagem alargada $\mathcal{L}'$) tal que:

- cada ϕ_i é da forma $\forall x_1 \ldots \forall x_n (\lambda_1 \vee \ldots \vee \lambda_n)$, onde cada λ_i é um literal
- a fórmula ϕ é satisfazível se e só se Φ é satisfazível

Algoritmo de Skolemização

1. Converter ϕ em forma normal prenexa $Q_1 x_1 Q_2 x_2 \ldots Q_n x_n \psi$ onde cada Q_i é ou $\forall$ ou $\exists$ e ψ não tem quantificadores (e denomina-se a *matriz*).
2. Para $1 \leq i \leq n$ se Q_i é um quantificador existencial $\exists$ e $x_{i_1}, \ldots x_{i_{m_i}}$ as variáveis quantificadas universalmente de índice menor que i então substitui-se:

   $$\exists x_i \theta \text{ por } \theta[f_i(x_{i_1}, \ldots x_{i_{m_i}})/x_i]$$

   onde f_i é um novo símbolo funcional de aridade m_i.

Algoritmo de Skolemização

3. Converter a matriz da fórmula resultante para forma normal conjuntiva aplicando sucessivamente as seguintes transformações:

   $$\begin{array}{lll} \neg\neg\phi & \longrightarrow & \phi \\ \neg(\phi \wedge \psi) & \longrightarrow & \neg\phi \vee \neg\psi \\ \neg(\phi \vee \psi) & \longrightarrow & \neg\phi \wedge \neg\psi \\ \phi \vee (\psi \wedge \theta) & \longrightarrow & (\phi \vee \psi) \wedge (\phi \vee \theta) \end{array}$$

   :

4. Aplicar a seguinte transformação:

   $$\forall x(\phi \wedge \psi) \quad \longrightarrow \quad \forall x \phi \wedge \forall x \psi$$

Exercício 22.1. *Aplica o algoritmo à seguinte fórmula:*

$$\forall x(\forall y(P(y) \rightarrow R(y,x)) \rightarrow Q(x))$$

$\diamond$

Resolução 22.1

Para forma normal prenexa basta passar para fora os quantificadores:

$$\forall x \forall y((P(y) \to R(y,x)) \to Q(x))$$

Como não tem quantificadores existênciais basta, converter a matriz para forma normal conjuntiva:

$$\forall x \forall y(\neg(\neg P(y) \ \lor \ R(y,x)) \ \lor \ Q(x)) \tag{1}$$
$$\forall x \forall y((P(y) \ \land \ \neg R(y,x)) \ \lor \ Q(x)) \tag{2}$$
$$\forall x \forall y((P(y) \ \lor \ Q(x)) \ \land \ (\neg R(y,x) \ \lor \ Q(x))) \tag{3}$$
$$\forall x \forall y((P(y) \ \lor \ Q(x))) \ \land \ \forall x \forall y(\neg R(y,x) \ \lor \ Q(x)) \tag{4}$$

Resolução para cláusulas fechadas

A regra da resolução para cláusulas fechadas é praticamente igual à da lógica proposicional.

Sejam $C \cup \{l\}$ e $C' \cup \{\neg l\}$ duas cláusulas fechadas.

Regra de inferência da Resolução

$$\frac{C \cup \{l\} \qquad C' \cup \{\neg l\}}{C \cup C'}$$

$C \cup C'$ é a **resolvente** de $C \cup \{l\}$ e $C' \cup \{\neg l\}$,

Exemplo 22.1. *Sendo* $\{P(f(a), g(c)), Q(a,c)\}$ *e* $\{\neg P(f(a), g(c)), Q(f(a), g(c))\}$. *Então, uma resolvente é*

$$\{Q(a,c), Q(f(a), g(c))\}$$