

Predikátová logika

prvního řádu

Predikát

Predikát je n -ární relace

- vyjadřuje vlastnosti objektů a vztahy mezi objekty
- z jednoduchého výroku vznikne vypuštěním alespoň jednoho jména objektu (individua)
- značíme velkými písmeny, např. predikáty P, Q, R, P_1, P_2
- unární predikáty
 - vyjadřují vlastnosti
- binární, ternární, ..., n -ární
 - vyjadřují vztahy mezi dvojicemi, trojicemi, ..., n -ticemi objektů.
- nulární predikáty
 - představují původní výroky (tzn. bez vypuštěných jmen objektů) ve výrokové logice
 - v predikátové logice se označují jako **sentence**

Konstanty, proměnné, funkce

Konstanty

- reprezentují jména objektů
- jedná se o prvky předem specifikované množiny hodnot – **domény**

Proměnné

- zastupují jména objektů
- mohou nabývat libovolných hodnot z dané domény

n -ární predikáty lze chápat jako množiny takových n -tic konstant, pro které je predikát splněn

Funkce

- reprezentují složená jména objektů
- konstanty jsou nulární funkce
- příklad: Nechť funkce $f(x,y)$ reprezentuje sčítání. Pak $f(1,2)$ (stejně jako $f(2,1), f(0,3)$) jsou možná složená jména pro konstantu 3.

Termy

Termy

- výrazy složené pouze z funkčních symbolů, konstant a proměnných
 - příklad termu: $f(x, g(y, h(x, y), 1), z)$
 - termy mohou nabývat hodnot v rámci dané domény (resp. její podmnožiny označované jako obor hodnot funkce)
1. každá proměnná a každá konstanta je term
 2. je-li f n -ární funkční symbol a t_1, \dots, t_n jsou termy, pak $f(t_1, \dots, t_n)$ je term
 3. nic jiného není term

Termy bez proměnných označujeme jako **uzavřené termy**

Kvantifikátory

Složené predikáty lze kromě běžných pravdivostních spojek vytvářet pomocí *kvantifikátorů*.

- *univerzální (obecný) kvantifikátor* \forall :
 - $\forall xP(x)$ – pro každý prvek x domény platí $P(x)$
- *existenční kvantifikátor* \exists :
 - $\exists xP(x)$ – pro některé prvky x domény platí $P(x)$ (resp. existuje alespoň jeden prvek x domény, pro který platí $P(x)$)

Univerzální kvantifikátor je zobecněním konjunkce a existenční je zobecněním disjunkce pro nekonečné domény

Formalizace jazyka predikátové logiky

Jazyk predikátové logiky: abeceda + pravidla pro správné tvoření termů a formulí

Abeceda:

- proměnné (nespecifikovaná jména objektů): x, y, z, x_1, x_2, \dots
- konstanty (vlastní jména objektů): a, b, c, a_1, a_2, \dots
- symboly pro spojky: $\neg, \vee, \wedge, \Rightarrow, \Leftrightarrow$
- symboly pro kvantifikátory: \forall, \exists
- n -ární funkční symboly (složená jména objektů): f, g, h, f_1, f_2, \dots
- n -ární predikátové symboly: P, Q, R, P_1, P_2, \dots
- pomocné symboly: $(,)$

Formule

Atomická formule

1. Je-li P n -ární predikátový symbol a t_1, \dots, t_n jsou termy, pak $P(t_1, \dots, t_n)$ je atomická formule
2. Jsou-li t_1 a t_2 termy, pak $t_1 = t_2$ je atomická formule
3. Nic jiného není atomická formule

Formule

1. Každá atomická formule je formule
2. Je-li A formule, pak $\neg A$ je formule
3. Jsou-li A, B formule, pak $(A \vee B)$, $(A \wedge B)$, $(A \Rightarrow B)$, $(A \Leftrightarrow B)$ jsou formule
4. Je-li x proměnná a A formule, pak $(\forall x A)$ a $(\exists x A)$ jsou formule
5. Nic jiného není formule

Vázaný a volný výskyt proměnných

Podformule formule A je libovolná spojitá podčást A , která je sama formulí.

Výskyt proměnné x ve formuli A je **vázaný**, pokud existuje podformule B formule A , která obsahuje tento výskyt x a začíná $\forall x$, resp. $\exists x$. Výskyt proměnné je **volný**, není-li vázaný.

Proměnná x se *volně vyskytuje* v A , má-li tam alespoň jeden volný výskyt

Sentence (uzavřená formule) predikátové logiky je formule bez volných výskytů proměnných (všechny výskyty všech proměnných jsou vázané)

Otevřená formule je formule bez kvantifikátorů

Substituce proměnných

Proměnné, za které lze dosadit (udělit jim hodnotu, provést substituci), jsou pouze *volné* proměnné

Term t je substituovatelný za proměnnou x ve formuli A , pokud pro každou proměnnou y obsaženou v t neobsahuje žádná podformule A tvaru $\forall yB$, $\exists yB$ volný výskyt proměnné x

Je-li t substituovatelný za x v A , označíme $A(x/t)$ výraz, který vznikne z A nahrazením každého volného výskytu x termem t

Příklad: Ve formuli $A = \exists xP(x,y)$ lze provést například tyto substituce:

$$A(y/z) = \exists xP(x,z); \quad A(y/2) = \exists xP(x,2); \quad A(y/f(z,z)) = \exists xP(x,f(z,z));$$

Není však možné substituovat např. takto: $A(y/f(x,x)) = \exists xP(x,f(x,x))$, protože by došlo k nežádoucí vazbě

Interpretace jazyka

- Interpretace (realizace) jazyka predikátové logiky je struktura I složená z
 - libovolné neprázdné množiny D (domény, oboru interpretace)
 - zobrazení $I(f): D^n \rightarrow D$ pro každý n -ární funkční symbol f , $n \geq 0$
 - n -ární relace $I(P) \subseteq D^n$ pro každý n -ární predikátový symbol P , $n \geq 1$
- Příklad: Mějme jazyk s binárním predikátovým symbolem $P(x, y)$, binárním funkčním symbolem $f(x, y)$ a symboly pro konstanty $a, a_1, a_2, \dots, b_1, b_2, \dots$, který chceme interpretovat jako celí čísla s $> a +$.
 - D je množina celých čísel,
 - $I(a) = 0, I(a_1) = 1, I(a_2) = 2, \dots, I(b_1) = -1, I(b_2) = -2, \dots$
 - $I(f) = +$ (např. $I(f)(4, -2) = 2$)
 - $I(P) = >$ (např. $(2, -1) \in I(P)$)

Interpretace proměnných a termů

Interpretace volných proměnných spočívá v jejich ohodnocení, což je libovolné zobrazení V (valuace) z množiny všech proměnných do D

Ohodnocení, které přiřazuje proměnné x prvek $d \in D$ a na ostatních proměnných splývá s valuací V , označíme $V[x/d]$

Hodnotou termu t v interpretaci I a valuaci V je

- $V(t)$, je-li term t proměnná
- Hodnota funkce $I(f)$ pro argumenty t_1, \dots, t_n , je-li $t = f(t_1, \dots, t_n)$

Pravdivost formulí a jejich klasifikace

Formuli A nazveme pravdivou v interpretaci I , je-li splňována I pro libovolnou valuaci, píšeme $\models_I A$

Pravdivost formule záleží pouze na valuaci volných proměnných, které se v ní vyskytují

Pravdivost sentence (uzavřené formule) nezávisí na valuaci vůbec

Formule A *predikátové logiky se nazývá*

- *tautologie*, je-li pravdivá pro každou interpretaci (tj. pro každou I platí $\models_I A$), značíme $\models A$
- *splnitelná*, pokud existuje alespoň jedna interpretace a valuace, které ji splňují
- *kontradikce*, je-li $\neg A$ tautologie (tj. $\models \neg A$)

Tautologie predikátové logiky

Získané z tautologií výrokové logiky:

- Nahradíme-li v tautologii výrokové logiky korektně všechny výrokové symboly formulami predikátové logiky, získáme tautologii predikátové logiky
- Příklad:
V tautologii výrokové logiky $\models p \vee \neg p$ nahradíme p formulí $\forall xP(x)$, získáme tak tautologii predikátové logiky $\models \forall xP(x) \vee \neg \forall xP(x)$

Prenexová normální forma (pnf)

- Cílem je převést libovolnou (uzavřenou) formuli do tvaru, v němž jsou všechny kvantifikátory na začátku a následuje otevřené (tzn. bez kvantifikátorů) jádro v nkf (ndf)

$$Qx_1 \dots Qx_n ((A_{1_1} \vee \dots \vee A_{1_{l_1}}) \wedge (A_{2_1} \vee \dots \vee A_{2_{l_2}}) \wedge \dots \\ \wedge (A_{m_1} \vee \dots \vee A_{m_{l_m}}))$$

- Příklad: $\forall x \forall y \exists z \forall w ((P(x, y) \vee \neg Q(z)) \wedge (R(x, w) \vee R(y, w)))$
- **Věta:** Pro každou formuli existuje ekvivalentní formule v konjunktivní (disjunktivní) prenexové normální formě

Pnf: algoritmus převodu

1. Eliminovat zbytečné kvantifikátory
2. Přejmenovat korektně proměnné tak, aby u každého kvantifikátoru byla jiná proměnná
3. Eliminovat všechny spojky různé od \neg , \vee a \wedge
4. Přesunout negaci dovnitř, je-li potřeba:
 - $\neg \forall x A$ nahradit $\exists x \neg A$
 - $\neg (A \wedge B)$ nahradit $\neg A \vee \neg B$ apod.
5. Přesunout kvantifikátory doleva ($\diamond \in \{\vee, \wedge\}$, $Q \in \{\forall, \exists\}$):
 - $A \diamond Qx B$ nahradit $Qx(A \diamond B)$
 - $QxA \diamond B$ nahradit $Qx(A \diamond B)$
6. Použít distributivní zákony k převodu jádra do nkf (ndf):
 - $A \vee (B \wedge C)$ nahradit $(A \vee B) \wedge (A \vee C)$
 - $(A \wedge B) \vee C$ nahradit $(A \vee C) \wedge (B \vee C)$

Skolemizace

- Převod formulí na formule bez existenčních kvantifikátorů v jazyce, který je rozšířen o tzv. Skolemovy funkce
- Idea převodu: formuli $\forall x_1 \dots \forall x_n \exists y P(x_1, \dots, x_n, y)$ převedeme na $\forall x_1 \dots \forall x_n P(x_1, \dots, x_n, f(x_1, \dots, x_n))$
- Příklad: Mějme celá čísla s +. Formuli $\forall x \exists y (x + y = 0)$ převedeme na $\forall x (x + f(x) = 0)$. Interpretace f – unární funkce, která pro daný argument vrátí opačné číslo.
- Skolemova normální forma je prenexová normální forma pouze s univerzálními kvantifikátory

Algoritmus převodu do Skolemovy nf

1. Převést formuli do (konjunktivní) prenexové normální formy
2. Provést Skolemizaci: odstranit všechny existenční kvantifikátory a nahradit jimi vázané proměnné pomocnými Skolemovými funkcemi

- Příklad 1: Převeďte do Skolemovy nf formuli

$$\forall x \exists y \neg (P(x, y)) \Rightarrow \forall z R(y) \vee \neg \exists x Q(x)$$

$$1. \forall x_1 \exists y \forall x_2 ((P(x_1, y) \vee \neg Q(x_2)) \wedge (\neg R(y) \vee \neg Q(x_2)))$$

$$2. \forall x_1 \forall x_2 ((P(x_1, f(x_1)) \vee \neg Q(x_2)) \wedge (\neg R(f(x_1)) \vee \neg Q(x_2)))$$

- Příklad 2: Převeďte do Skolemovy nf následující formuli v pnf

$$\forall x \exists y \forall z \exists w (P(x, y) \vee \neg Q(z, w))$$

$$\forall x \forall z (P(x, f_1(x)) \vee \neg Q(z, f_2(x, z)))$$