Kompleksni podatkovni tipi in stavčna logika

Ljupčo Todorovski

Univerza v Ljubljani, Fakulteta za matematiko in fiziko
Institut Jožef Stefan, Odsek za tehnologije znanja (E8)

April 2023

Vhodni podatki: tabeli objav in referenc

Tabela objava

Avtor	Naslov
quinlan	learning logical definitions from relations
lloyd	foundations of lp
lloyd	logic for learning
russell	ai a modern approach
norvig	ai a modern approach

Tabela referenca

Naslov_1	Naslov_2
logic for learning	foundations of lp
logic for learning	learning logical definitions from relations
logic for learning	ai a modern approach
ai a modern approach	foundations of lp

Ciljni podatki: tabela citatov

Tabela citat

$Avtor_1$	$Avtor_2$
lloyd	lloyd
lloyd	quinlan
lloyd	russel
russel	lloyd
norvig	lloyd
⋮	

Naloga učenja

- Iščemo model (definicijo) ciljne tabele *Citat*
- Na osnovi tabel *Objava* in *Referenca*

Posebnosti strojnega učenja

Podatki niso zapisani v eni tabeli

- Boolova ciljna spremenljivka: podatki iz tabele *Citat*
- Atributi: podatki iz tabel *Objava* in *Referenca*

vhodne spremenljivke

Vrednost ciljne spremenljivke

- *True*: za par avtorjev (a_1, a_2), kjer a_1 citira a_2
- *False*: za vse druge pare avtorjev

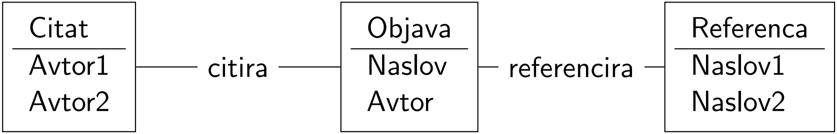
Rezultat učenja: pravilo IF-THEN

- IF
  - a_1 je avtor o_1 IN
  - o_1 vsebuje referenco na o_2 IN
  - a_2 je avtor o_2
- THEN a_1 citira a_2

$$Objava(a_1, o_1) \land Referenca(o_1, o_2) \land Objava(a_2, o_2) \implies Citat(a_1, a_2)$$

Primer BIB: Podatki v podatkovni bazi

E-R model podatkovne baze



Primer BIB: Rezultat učenja je poizvedba

Definicija tabele *Citat* s poizvedbo SQL

to je način kako iz tabele objava in referenca zgradim tabelo citat.

```
SELECT t1.Avtor, t2.Avtor
      FROM Objava t1, Objava t2, Referenca t3
      WHERE t1.Naslov = t3.Naslov1 AND t2.Naslov = t3.Naslov2
```

Primer BIB: Podatki kot predikati in dejstva

predikat OBJAVA ki poveže avtorja z njegovim naslovom

Predikat *objava*/2 in dejstva (primeri)

objava(quinlan, 'learning logical definitions from relations').
objava(lloyd, 'foundations of lp').
⋮

Predikat *referenca*/2 in dejstva (primeri)

referenca('logic for learning', 'foundations of lp').
⋮
referenca('ilp theory and methods', 'foundations of lp').

Podobno tudi *citat*/2.

Primer BIB: Rezultat učenja je logični stavek

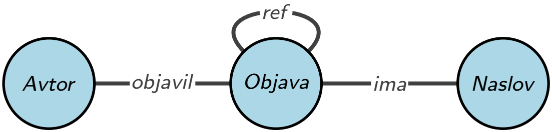
Definicija predikata *citat* s stavkom

$citat(A_1, A_2) \leftarrow objava(A_1, O_1), referenca(O_1, O_2), objava(A_2, O_2).$

Prvi del predavanj v tem poglavju: strojno učenje logičnih stavkov.

Primer BIB: podatki predstavljajo omrežje

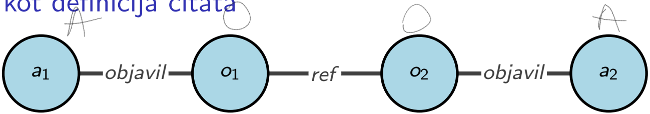
Shema omrežja



Primer BIB: rezultat učenja je meta sprehod

sprehod iz avtorja v objavo pa v drugo objavo pa v avtorja.

Meta sprehod AOOA kot definicija citata

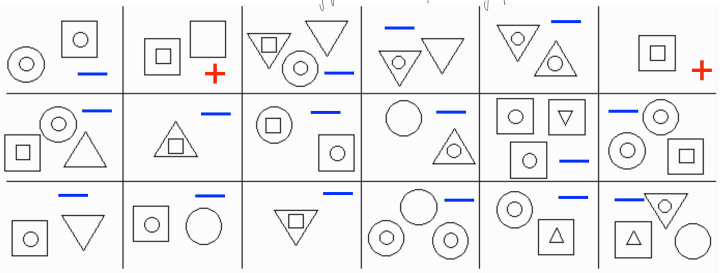


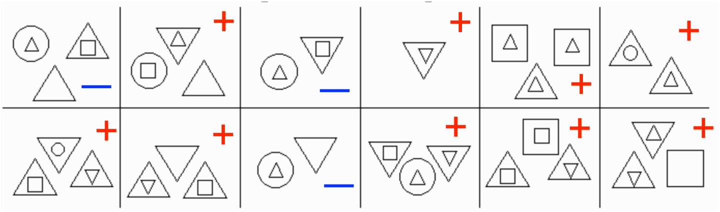
Drugi del predavanj v tem poglavju: strojno učenje v omrežjih/grafih.

Pozitivni in negativni primeri

3 različni geom. liki. En drugega lahko vsebujejo. Eni primeri so ⊕ eni pa ⊖ (to vidmo)

Kaj je razlika med poz. in neg. primeri?





Štirje predikati

kako to sliko spravit v predikate?

Trije enomestni predikati za oblike

- krog/1
- kvadrat/1
- trikotnik/1

geo. oblike ki jih imamo

ko neki lik vsebuje neki lik

Dvomesten predikat vsebuje/2.

kako primer 

predstavit s temi predikati?

krog(k1), krog(k2), krog(k3)
kvadrat(v1), vsebuje(k1,k2),
vsebuje(v1,k3)

(PREZENTACIJA BONGARDOVEGA DIAGRAMA S PREDIKATI)

Predikatni opis primerov

Prvi negativni primer	krog(k1), krog(k2), krog(k3), kvadrat(v1), vsebuje(k1, k2), vsebuje(v1, k3)
Prvi pozitivni primer	kvadrat(v1), kvadrat(v2), kvadrat(v3), vsebuje(v1, v2)
Tretji pozitivni primer	krog(k1), kvadrat(v1), trikotnik(t1), trikotnik(t2), trikotnik(t3), vsebuje(k1, v1), vsebuje(t1, t2)

Razlika od običajnega strojnega učenja

Primeri

- So množice namesto vektorjev oz. p-teric
- Vsebujejo strukturirane izraze (predikate)

Model za razvrščanje: Pozitivni primer

- Ima vsaj en trikotnik vsebovan v drugem trikotniku
- Sicer, če nima trikotnika, ne sme imeti niti kroga (pol je sestavljen iz samih □)

Kako formalno zapišemo tak model? en način je v obliki odločitvenega drevesa (glej naslednjo stran)

take diagrame je težko pravilno brat

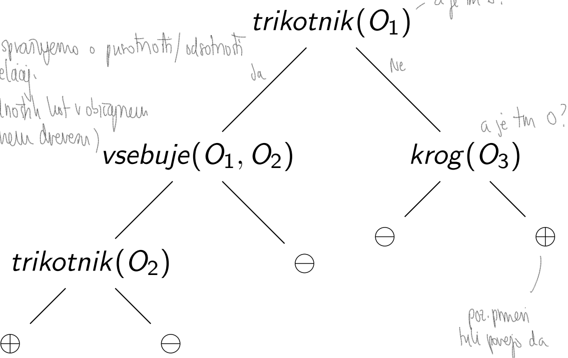
Relacijsko, lahko tudi predikatno drevo

Rešitev zgornje tike. Ali je primer ⊕ ali ⊖.

PREDIKATNO DREVO (ODLOČITVENO DREVO.)

ker se sprašujemo o prisotnosti / odsotnosti nekih relacij.

(ne po vrednosti kot v običajnem odločitvenem drevesu)



poz. primer ker je △.

poz. primeri tuli povejo da ni △ pa je ○.

Kakšna je semantika relacijskega drevesa?

(Ne tako zelo) kratek izlet v logiko prvega reda (stavčno logiko).

Elementi sintakse stavčne logike (*clausal logic*)

vedno jih lahko vrednotimo na True pa False

TUKI TABELA SNEMAT

pišemo jih z malo začetnico da jih ne mešamo s spremenljivkami

kompliciran izraz (ker ima oklepaje, vejice)

sami po sebi nimajo vrednosti (logične)

- Izrazi, *terms*
- Atomi, *atoms*
- Stavki, *clauses*

IZRAZI	ATOMI	STAVKI
enostaven izraz → k_1	krog$(k_1)^T$ True	
oseba(jupšo, marta)	kvadrat$(v_1)^T$ False	
oseba(k_1, jupšo)	kvadrat(k_3) ⊥	
	kvadrat(O_1) ↑ sprem!	

lahko tut kompliciranih oblik (veliko funkcije)

tvorimo na njih naše logične izraze, saj za njih lahko povemo a je True al False. Atomi so načeloma predikati

ampak tut spremenljivke so lahko izrazi

(spremenljivke niso ground term, lahko so če jim dodelimo vrednost)

ta izraz sam navaja dejstvo o neki osebi (ni predikat)

Pomembna razlika med izrazi in atomi

- Izrazom ne prirejamo resničnostih vrednosti
- Atomom lahko priredimo resničnostno vrednost

če izrazi imajo fiksno vrednost (ground terms)

zaplet ker smo predikatu kvadrat dal spremenljivko!

kvadrat$(O_1)^T$ če $O_1 = v_1$

ničeopaje za O_1 so sam k_1, k_2, k_3, v_1 (iz prejšnjega primera)

neg. primer k smo zgori pogledali

ta je T na 3 možne načine $O_3=k_1$, $O_3=k_2$, $O_3=k_3$ (dost da je sam en način da je true)

trikotnik(O_2) ⊥ vedno, ker ne obstaja vrednost spremenljivke O_2 da bi (lahko dal vrednost)

krog$(O_3)^T$

Izrazi: Konstante in sestavljeni izrazi

Se nanašajo na objekte v opazovani domeni.

Konstante: nizi znakov z malo začetnico

Primera: *lloyd* (BIB), *t1* (BONGARD)

Spremenljivke: nizi znakov z veliko začetnico

- Lahko jim priredimo poljubni izraz kot vrednost
- Primeri: *Avtor*, *A*, *Oblika*, *O*

Strukture ali sestavljeni izrazi, *compound terms*

- Funkcija f/n, kjer n označuje število argumentov f
- Primer *oseba/2*: *oseba(lloyd, moški)*

pomeni, da poveže dve stvari medseboj

Opredeljenost izrazov

Popolnoma opredeljen izraz, *ground term*

Nanaša se na en objekt iz opazovane domene. Je lahko

- Konstanta
- Sestavljeni izraz brez spremenljivk

Primeri

- *lloyd* in *oseba(lloyd, moški)* sta popolnoma opredeljena izraza
- *Oseba* in *oseba(Oseba, ženska)* nista, saj se lahko nanašata na katerokoli osebo *Oseba* (objekt iz opazovane domene)

Atomi in literali

Predikati

- Predikat p/n, kje n označuje število njegovih argumentov
- Predstavlja relacijo med izrazi ali pa lastnost izraza
- Primer: *objava/2: objava(lloyd, foundations_of_lp)*
- Primer: *trikotnik/1: trikotnik(t1)*

Razlika med atomi (predikati) in (sestavljenimi) izrazi

- Strukturirani izrazi se nanašajo na objekte v domeni
- Atomi (predikati) se nanašajo na relacije med objekti
- Za atome lahko ugotavljamo resničnostno vrednost

Literal je lahko atom ali logična negacija atoma.

Stavki in teorija

ločimo z implikacijo

Stavke tvorimo iz atomov in literalov z implikacijo

nekaj literalov na desno stran stavka in jih ločimo z ,

ZAPIS LOGIČNE FORMULE

$$h_1; h_2; \ldots; h_m \leftarrow b_1, b_2, \ldots, b_n$$

na levi strani stavka dano skupi nekaj literalov pa jih ločimo s ;

- h_j in b_i so atomi oz. literali
- Glava (*head*) stavka je disjunkcija $h = h_1 \vee h_2 \vee \ldots \vee h_m$ (h1 ali h2 ali... ali hm

  (leva stran stavka)
- Telo (*body*) stavka je konjunkcija $b = b_1 \wedge b_2 \wedge \ldots \wedge b_n$

  ↳ (b1 in b2 in .. in bn)
- Logični pomen stavka je $b \implies h$

konjunkcija vseh literalov, ki se pojavijo v telesu

Če upoštevamo, da je $p \implies q$ isto kot $\neg p \vee q$

INTERPRETACIJA S KONJUNKCIJO

$$\neg b_1 \vee \neg b_2 \vee \ldots \vee \neg b_n \vee h_1 \vee h_2 \vee \ldots \vee h_m$$

Teorija (tudi logični program) je množica stavkov.

Demorganov zakon

negacija konjunkcij je disjunkcija

Primeri stavkov brez spremenljivk (1)

logične izjave:

srečna ali žalostna je ∀ oseba

$$srecna; zalostna \leftarrow oseba$$

Osebe so lahko srečne ali žalostne.

$$\neg oseba \vee srecna \vee zalostna$$

ne oseba ali srečna ali žalostna

- Oseba, ki ni srečna, je žalostna. Oseba, ki ni žalostna, je srečna.
- Pozor: oseba je lahko hkrati srečna in žalostna.
- Kako napišemo stavek, ki tega ne omogoča?

Primeri stavkov brez spremenljivk (2)

$$\leftarrow oseba, srecna, zalostna$$

Ni osebe, ki bi bila srečna in žalostna hkrati.

$$\neg oseba \vee \neg srecna \vee \neg zalostna$$

Primeri stavkov s spremenljivkami (1)

21min

$$tail\_of(t(X), X) \leftarrow mouse(X)$$

Vsaka miška ima rep.

$$\forall X \neg mouse(X) \vee \underbrace{tail\_of(t(X), X)}_{\text{funkcija k miški določi rep}}$$

- Spremenljivka X vezana z univerzalnim kvantifikatorjem
- Izraz $t(X)$ pove, da je rep odvisen od miške
- Alternativen zapis $\forall X \exists T \neg mouse(X) \vee tail\_of(T, X)$
- Alternativni zapis nam pomaga popolnoma razumeti semantiko stavka

Primeri stavkov s spremenljivkami (2)

$$\leftarrow \mathit{trikotnik}(X), \mathit{trikotnik}(Y), \mathit{vsebuje}(X, Y)$$

stavčna logika prevedena v običajno? logiko

Ni trikotnika, ki bi vseboval (drug) trikotnik.

$$\forall X \forall Y \underbrace{\neg \mathit{trikotnik}(X) \vee \neg \mathit{trikotnik}(Y) \vee \neg \mathit{vsebuje}(X, Y)}_{\text{demorgan}}$$

- Pri interpretaciji si spet lahko pomagamo z alternativnimi zapisani
- Prvi zapis: $\forall X \forall Y \neg(\mathit{trikotnik}(X) \wedge \mathit{trikotnik}(Y) \wedge \mathit{vsebuje}(X, Y))$
- Drugi zapis: $\neg \exists X \exists Y\ \mathit{trikotnik}(X) \wedge \mathit{trikotnik}(Y) \wedge \mathit{vsebuje}(X, Y)$

ni trikotnika, ki bi vseboval drug trikotnik

Posebni razredi stavkov

- Dejstva: prazno telo in en atom v glavi, $m = 1$, $n = 0$
- Definitni (določeni) stavki: en atom v glavi, $m = 1$, n poljuben
- Zanikanja ali poizvedbe: prazna glava, $m = 0$, n poljuben
- Hornovi stavki: največ en atom v glavi $m \leq 1$, n poljuben

Primeri posebnih stavkov

ni telesa, zato lahko napišemo

Dejstvo: $trikotnik(t1) \leftarrow$

$$\neg\top \vee trikotnik(t1) \equiv \bot \vee trikotnik(t1) \equiv trikotnik(t1)$$

Zato tudi krajši, pogosto uporabljen zapis $\boxed{trikotnik(t1).}$

rečemo da poizvedba ne uspe če je trikotnik uspešen

Zanikanje oz. poizvedba, primera: $\leftarrow trikotnik(t1)$ in $\leftarrow trikotnik(O)$

$$\neg trikotnik(t1) \vee \bot \equiv \neg trikotnik(t1)$$

$$\forall O\, \neg trikotnik(O) \vee \bot \equiv \forall O\, \neg trikotnik(O) \equiv \neg\exists O\, trikotnik(O)$$

Hierarhija tipov stavčne logike

Propozicijska logika

Izrazi so lahko le resnični ali neresnični: ni spremenljivk.

Relacijska logika

Objekte lahko naslavljamo s konstantami ali spremenljivkami.

Polna logika

Lahko imamo strukturirane izraze.

Definitna (določena) logika

Le defintini (določeni) stavki: ni disjunkcije v glavi stavka.

Zamenjava (*substitution*) in instanca stavka

Zamenjava preslika spremenljivke v izraze

Od neopredeljenih proti popolnoma opredeljenimi izrazi (stavki).

Instanca stavka

- Ko zamenjavo apliciramo na stavek, dobimo njegovo *instanco*
- Če instanca ne vsebuje spremenljivk, dobimo popolnoma določen stavek (popolnoma določeno instanco)

Vse instance stavka so njegove logične posledice.

Primer zamenjave

Stavek c pred zamenjavo

$$tail\_of(t(X), X) \leftarrow mouse(X).$$

Zamenjava θ

$$\theta : X = mickey$$

Instanca stavka c_θ

mickey ima rep t(mickey)

$$tail\_of(t(mickey), mickey) \leftarrow mouse(mickey).$$

Je popolnoma določena instanca, saj ne vsebuje spremenljivk.

Algoritem, ki s pomočjo zamenjav preveri pravilnost nekega izraza

Herbradnove interpretacije

Herbradnov univerzum

Množica vseh določenih izrazov (konstant).

Herbrandova baza

Množica vseh določenih atomov, t.j., literalov brez spremenljivk.

Herbrandova interpretacija

Množica *resničnih* določenih atomov.

Resničnost stavka in Herbrandov model

vedno vezana na Herbrandov model, ker rabmo univerzum ki nam da možnosti zax.

Resničnost stavka

- Stavek je *neresničen za podano interpretacijo*, če so vsi literali *v telesu resnični* in vsi literali *v glavi neresnični*.
- Sicer je stavek *resničen za podano interpretacijo*

Herbradnov model stavka

Interpretacija, kjer so vse določene instance stavka resnične.

Diagrami BONGARD

Podana sta

- Herbrandov univerzum *k1, v1, t1, t2, t3* ter
- Herbrandova interpretacija *krog(k1), kvadrat(v1), trikotnik(t1), trikotnik(t2), trikotnik(t3), vsebuje(k1, v1), vsebuje(t1, t2)*

ni trikotnika, ki bi bil vsebovan v trikotniku

Preveri veljavnost stavka (oz. uspešnost poizvedbe)

$$\leftarrow trikotnik(X), trikotnik(Y), vsebuje(X, Y).$$

Poizvedba ima obliko stavka zanikanja zaradi izpeljave s protislovjem.

Diagrami BONGARD: Logična izpeljava s protislovjem

$\leftarrow$ *trikotnik*(X), *trikotnik*(Y), *vsebuje*(X, Y).

Zamenjava $X = t1$

$\leftarrow$ *trikotnik*$(t1)$, *trikotnik*(Y), *vsebuje*$(t1, Y)$.

Zamenjava $Y = t2$

$\leftarrow$ *trikotnik*$(t1)$, *trikotnik*$(t2)$, *vsebuje*$(t1, t2)$.

$$\neg trikotnik(t1) \vee \neg trikotnik(t2) \vee \neg vsebuje(t1, t2)$$

Protislovje s Herbradnovo interpretacijo na prejšnji prosojnici.

Stavek torej ni resničen, a to vendar pomeni, da je poizvedba uspešna.

Semantika rezultata poizvedbe

Uspešna poizvedba

- Ob zamenjavi $\theta = \{X = t1, Y = t2\}$
- In je torej rezultat poizvedbe $X = t1, Y = t2$
- Pozor: rešitev poizvedbe je lahko več!

(na slajdu tegon je bla sam ena, v drugih primerih pa bi jih blo lahko več)

Pomen uspeha poizvedbe v logiki prvega reda, glej prosojnico 25

$$\exists X, Y \; trikotnik(X) \wedge trikotnik(Y) \wedge vsebuje(X, Y)$$

Rečemo tudi, da poizvedba *pokriva* podano Herbradnovo interpretacijo.

Relacijska odločitvena drevesa: vozlišča

Uspešna poizvedba pomeni, da je bil stavek neresničen ?!

45 min

razlaga drevo!!

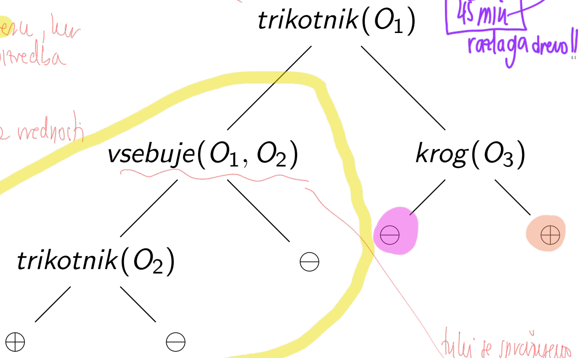
Kje bo definirana O_1?

Samo v levem poddrevesu, ker je bila samo tam poizvedba uspešna!

na desni pa nima nobene vrednosti

Kje je def. O_3?

na desnem pa O_3 ne $\exists$



tukaj je spraševano če obstaja objekt O_2, ki je vsebovan v O_1?

Pomen vozlišč

- Notranja vozlišča: poizvedbe, npr. $\leftarrow \mathit{trikotnik}(O_1)$
- Končna vozlišča: napovedi

45-59min

naša poizvedba:

$\leftarrow$ trikotnik(O_1)

preverimo če ta stavek velja

v Herbrandovem svetu

$\downarrow$ VELJA

v telesu torej to pomeni

$\forall O_1 \neg$ trikotnik(O_1)

če damo negacijo ven: $\boxed{\neg \exists O_1 \text{ trikotnik}(O_1)}$

✓ torej to velja $\Rightarrow$ Poizvedba NI USPEŠNA!!

gremo na desno vejo

Kaj pa ko velja?

naša poizvedba:

$\leftarrow$ krog(O_1) telo zato negiramo

$\forall O_1: \neg$ krog(O_1)

$\boxed{\neg \exists O_1: \text{krog}(O_1)}$ ne velja! (ni res) $\Rightarrow$ Poizvedba JE USPEŠNA!!

gremo na levo vejo

in najdemo kontra primere:

$O_1 = k_1$
$O_1 = k_2$
$O_1 = k_3$

$\}$ razlaga zakaj stavek ni res (to je rezultat poizvedbe)

Relacijska odločitvena drevesa: veje

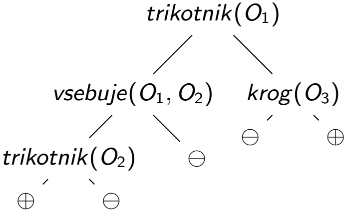
Leva veja

- Poizvedba v vozlišču uspešna
- Spremenljivke dobijo vrednosti iz rešitev poizvedbe
- Vrednosti spremenljivk *veljavne le v levi veji*

Desna veja

- Poizvedba v vozlišču ni uspešna
- Vrednosti spremenljivk iz poizvedbe neznane
- Zato tudi spremenljivke nimajo vrednosti v desni veji

Relacijska odločitvena drevesa: napovedi

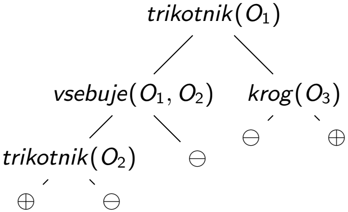


logični stavki, ki veljajo za ∀ od končnih vozlišč
tega drevesa

Končna vozlišča od leve proti desni

1. $\exists O_1, O_2 : trikotnik(O_1) \wedge vsebuje(O_1, O_2) \wedge trikotnik(O_2) \Rightarrow \oplus$
2. $\exists O_1, O_2 : trikotnik(O_1) \wedge vsebuje(O_1, O_2) \wedge \neg trikotnik(O_2) \Rightarrow \ominus$
3. $\exists O_1 : trikotnik(O_1) \wedge \nexists O_2 : vsebuje(O_1, O_2) \Rightarrow \ominus$
4. $\nexists O_1 : trikotnik(O_1) \wedge \exists O_3 : krog(O_3) \Rightarrow \ominus$
5. $\nexists O_1 : trikotnik(O_1) \wedge \nexists O_3 : krog(O_3) \Rightarrow \oplus$

Relacijska odločitvena drevesa: IF-THEN-ELSE



IF $trikotnik(O_1) \wedge vsebuje(O_1, O_2) \wedge trikotnik(O_2)$ THEN $\oplus$
ELIF $trikotnik(O_1) \wedge vsebuje(O_1, O_2)$ THEN $\ominus$
ELIF $trikotnik(O_1)$ THEN $\ominus$
ELIF $krog(O_3)$ THEN $\ominus$
ELSE $\oplus$

Konec ponavljanja LOGIKE PRVEGA REDA.