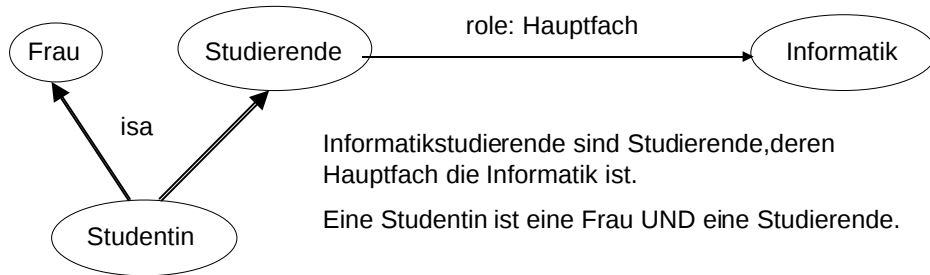


Termsubsumtionsformalismen (KL-ONE)

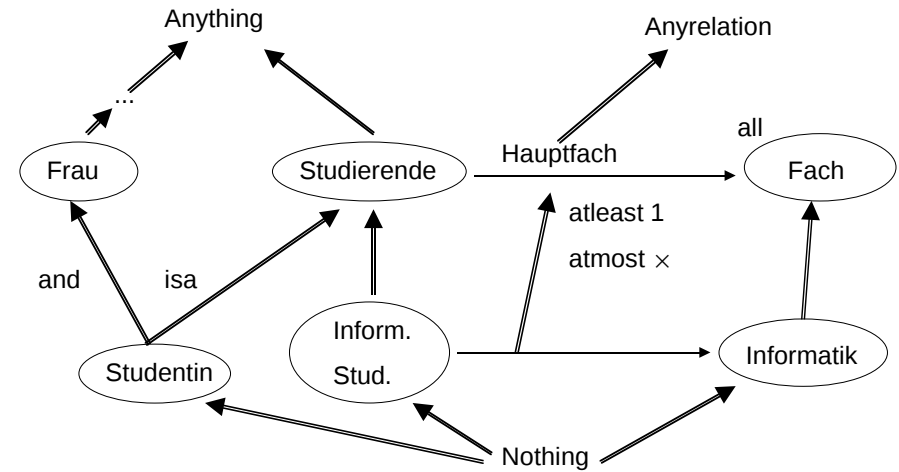
In epistemischen Primitiven Begriffe definieren. Teile der Signatur und der Interpretation sind vorgegeben. Einige Schlussfolgerungen sind vorbereitet.
≠ logische Ebene!



Termsubsumtion

1

T-Box für terminologisches Schließen



Studentin .:= (and Frau Studierende)
Informatikstudierende .:= (and Studierende (all Hauptfach Informatik))

Termsubsumtion

2

Syntax von Termsubsumtionsformalismen

```

<terminology> ::= {<term-introduction> | <restriction> }*
<term-introduction> ::= <concept-introduction> | <role-introduction>
<concept-introduction> ::= <atomic-concept> .:= <concept> |
    <atomic-concept> ≤ <concept> |
    <atomic-concept> ≤ ANYTHING
<role-introduction> ::= <atomic-role> .:= <role> |
    <atomic-role> ≤ <role> |
    <atomic-role> ≤ ANYRELATION
<concept> ::= <atomic-concept> |
    (and <concept>+) |
    (all <role> <concept>) |
    (atleast <number> <role>) |
    (atmost <number> <role>)
<role> ::= <atomic-role> |
    (androle <role>+)
<restriction> ::= (disjoint <atomic-concept> <atomic-concept>)
<number> ::= <non-negative-integer>
<atomic-role> ::= <identifier>
<atomic-concept> ::= <identifier>

```

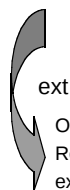
Wohlgeformte, aus epistemischen Primitiven zusammengesetzte Ausdrücke

Termsubsumtion

3

Semantik epistemischer Primitive und zusammengesetzter Ausdrücke

Knoten: Entscheidung über Begriffszugehörigkeit über notwendige Eigenschaften
d.h. $\forall x \ x \in \text{Begriff} \mid \dots$



ext

Objekte: a1, a2, a3, ...
Relationen (a1,b1),(a1,b2),(a1,b3),(a2,b1),...
ext

D
D x D

für alle $a \leq t$ sei $\text{ext}(t) \supseteq \text{ext}(a)$
für alle $a = t$ sei $\text{ext}(t) = \text{ext}(a)$
ext(ANYTHING) = D
ext(ANYRELATION) = D x D
 $\text{ext}(\text{and } c1 \dots cn) = \cap \text{ext}(ci)$ wobei i von 1 bis n
 $\text{ext}(\text{all } r \ c) = \{x \in D \mid \forall y : (x,y) \in \text{ext}(r) \rightarrow y \in \text{ext}(c)\}$
 $\text{ext}(\text{atleast } n \ r) = \{x \in D \mid \text{card}(\{y \in D \mid (x,y) \in \text{ext}(r)\}) \geq n\}$
 $\text{ext}(\text{atmost } n \ r) = \{x \in D \mid \text{card}(\{y \in D \mid (x,y) \in \text{ext}(r)\}) \leq n\}$
 $\text{ext}(\text{androle } r1 \dots rn) = \cap \text{ext}(ri)$ wobei i von 1 bis n
für alle (disjoint $c1 \ c2$) sei $\text{ext}(c1) \cap \text{ext}(c2) = \{\}$

Termsubsumtion

4

Klassifikation

- Gegeben eine TBox und eine neue Begriffsdefinition,
- finde den richtigen Platz für den neuen Begriff.

Der richtige Platz ist:

- der neue Begriff hängt unter seinem speziellsten Oberbegriff,
- der neue Begriff subsumiert alle unter ihm hängenden Begriffe.

Termsubsumtion

5

Subsumtion

- Logische Struktur (D, ext)
- t subsumiert t' , geschrieben $t \gg t'$, gdw.
 $\forall (D, \text{ext}) \text{ von } T: \text{ext}(t) \supseteq \text{ext}(t')$
- transitiv: $t \gg t', t' \gg t''$ dann $t \gg t''$
- reflexiv: $t \gg t$
- t äquivalent t' gdw. $\forall (D, \text{ext}) \text{ von } T: \text{ext}(t) = \text{ext}(t')$
- t inkohärent in T gdw. $\forall (D, \text{ext}) \text{ von } T: \text{ext}(t) = \{\}$
(unerfüllbar, z.B. (and (atleast 1 r) (atmost 0 r)))

Termsubsumtion

6

T

- Mit and und NOTHING ist T ein Halbverband, d.h.:
- Infimum (größter gemeinsamer Unterbegriff) gibt es immer!
- Die partielle Ordnung ist die Subsumtion $\gg$.
- T behandelt alle Modelle der Begriffsdefinitionen, NICHT bestimmte Mengen von Grundbeispielen!

Termsubsumtion

7

Klassifikationsalgorithmus

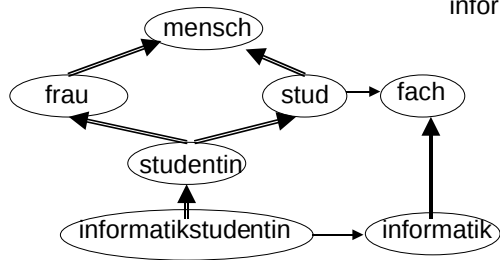
- Definitionen glätten, d.h. die Definitionen der Begriffe, die in einer Definition vorkommen, aufschreiben.
- bottom-up: von NOTHING ausgehend, bei jedem Begriff feststellen, ob der neue Begriff von dem Begriff subsumiert wird.
- Der Klassifikationsalgorithmus beruht also auf der Subsumtionsentscheidung.

Termsubsumtion

8

Beispiel

frau >> studentin?
ja, weil
studentin.=. (and s1 s2)
und
s1= frau



frau ≤ mensch
stud.=.(and mensch
(all studiert fach))
studentin.=. (and frau
(and mensch
(all studiert fach)))
informatikstudentin.=.
(and (and frau
(and mensch
(all studiert fach))
(all studiert informatik))

Termsubsumtion

9

Subsumtion

subsumes(t,u) ?

t	androle t ₁ ...t _n and t ₁ ...t _n	primitiv	all r _i c _i	atleast n _i r _i	almost n _i r _i
u					
NOTHING	wahr	wahr	wahr	wahr	wahr
primitiv	∀ _i 1 ≤ i ≤ n: subsumes (t _i , u)	t = u	falsch	falsch	falsch
all r _u c _u	∀ _i 1 ≤ i ≤ n: subsumes (t _i , u)	falsch	subsumes (c _i , c _u) ^ subsumes (r _u , r _i)	falsch	falsch
atmost 0 r _u	∀ _i 1 ≤ i ≤ n: subsumes (t _i , u)	falsch	subsumes (r _u , r _i)	falsch	subsumes (r _u , r _i) ^ 0 ≤ n _i
and u ₁ ...u _n androle u ₁ ...u _n	∀ _i 1 ≤ i ≤ n: subsumes (t _i , u)	∃ _i 1 ≤ i ≤ n: t = u _i	∃ _i 1 ≤ i ≤ n: subsumes (t, u _i)	∃ _i 1 ≤ i ≤ n: subsumes (t, u _i)	∃ _i 1 ≤ i ≤ n: subsumes (t, u _i)
atleast n _u r _u	∀ _i 1 ≤ i ≤ n: subsumes (t _i , u)	falsch	falsch	subsumes (r _i , r _u) ^ n _i ≤ n _u	falsch
atmost n _u r _u	∀ _i 1 ≤ i ≤ n: subsumes (t _i , u)	falsch	subsumes (r _u , r _i) bei n _u =0 sonst falsch	falsch	subsumes (r _u , r _i) ^ n _u ≤ n _i

Termsubsumtion

10

Subsumtionsalgorithmus

korrekt: wenn wahr für t >> u ausgegeben wird,
gilt ext(t) ⊇ ext(u).

vollständig: wenn ext(t) ⊇ ext(u) gilt,
wird wahr für t >> u ausgegeben.

Subsumtion ist entscheidbar, wenn Zyklen ausgeschlossen
sind.

Subsumtionsentscheidung ist polynomiell in der Länge der
Definitionen t und u, falls die Rollen primitiv bleiben.

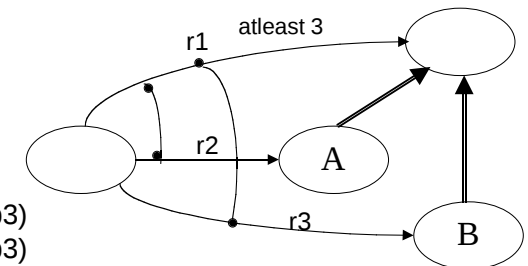
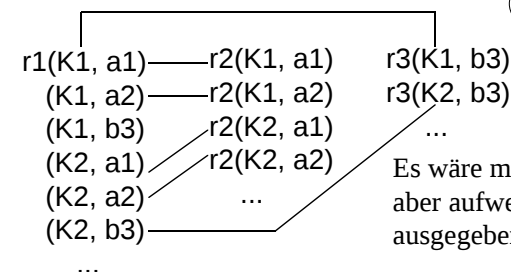
Termsubsumtion

11

Unvollständigkeit KL-One

t: (atleast 3 r₁)
u: (and (all (androle r1 r2) A)
(all (androle r1 r3) B)
(disjoint A B)
(atleast 2 (androle r1 r2))
(atleast 1 (androle r1 r3))

Schon androle macht die Subsumtion
unvollständig! disjoint wird von
subsumes nicht berücksichtigt.



Es wäre möglich, disjoint zu berücksichtigen,
aber aufwendig! Nicht immer, wenn wahr
ausgegeben werden müsste, wird es ausgegeben.

Termsubsumtion

12

Prinzip

Brachman, Levesque (87):

Handhabbarkeit vs. Ausdrucksfähigkeit eines
Wissensrepräsentationsformalismus’.

Man lebt mit polynomieller unvollständiger Subsumtion.

Termsubsumtion

13

T-Box

Das war die T-Box für

- terminologisches Schließen
- über allen Modellen ohne Grundbeispiele.

Jetzt kommt die A-Box für Assertionen, Grundformeln.

Termsubsumtion

14

A-Box

Syntax:

$\langle \text{world-description} \rangle ::= (\langle \text{obj descr} \rangle \mid \langle \text{rel descr} \rangle)$

$\langle \text{obj descr} \rangle ::= (\langle \text{atomic concept} \rangle \langle \text{obj} \rangle)$

$\langle \text{rel descr} \rangle ::= (\langle \text{atomic role} \rangle \langle \text{obj} \rangle \langle \text{obj} \rangle)$

$\mid (\langle \text{atomic role} \rangle \langle \text{obj} \rangle (\text{atleast} \langle \text{nr.} \rangle))$

$\mid (\langle \text{atomic role} \rangle \langle \text{obj} \rangle (\text{atmost} \langle \text{nr.} \rangle))$

Termsubsumtion

15

Semantik der A-Box

		D	Objekte
Funktion	int	2^D	Begriffe
		$2^{D \times D}$	Rollen

Interpretation (D,int) erfüllt δ

$\delta = (c \ o)$ ist erfüllt gwd. $\text{int}(o) \in \text{int}(c)$

$(r \ o \ p)$ $(\text{int}(o), \text{int}(p)) \in \text{int}(r)$

$(r \ o \ (\text{atleast } n))$ $\mid \{(\text{int}(o), x) \in \text{int}(r)\} \mid \geq n$

$(r \ o \ (\text{atmost } n))$ $\mid \{(\text{int}(o), x) \in \text{int}(r)\} \mid \leq n$

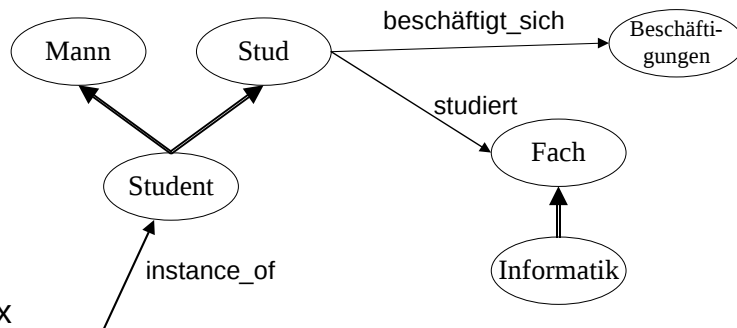
Wenn eine Interpretation int alle $\delta_i \in \text{A-Box}$ erfüllt, ist sie ein
Modell der A-Box. Wenn δ_j in allen Modellen der A-Box erfüllt
ist, gilt: $\text{A-Box} \mid = \delta_j$

Termsubsumtion

16

Beispiel

T-Box



A-Box

(Stud Hans)
 (beschäftigt_sich Hans (atmost 1))
 (beschäftigt_sich Hans Informatik)
 (Mann Hans)

Termsubsumtion

17

Beziehung zwischen T-Box und A-Box

Was studiert Hans?

Was ist der speziellste Begriff für Hans?
 (and Stud Mann)

Der allerspeziellste Begriff ist konstruierbar.

Vermutlich gibt es ihn nicht in T-Box.

HansBegriff .=. (and Stud
 Mann
 (all beschäftigt_sich Informatik)
 (atmost 1 beschäftigt_sich))

Klassifikation (HansBegriff)

subsumes(Student, HansBegriff)

Termsubsumtion

18

Hybride Inferenzen

- Inferenzen in der T-Box
Vererbung
- Inferenzen zwischen T-Box und A-Box:
 - Ist ein Objekt ein Grundbeispiel für einen Begriff?
covers (c, o)?
 - Welcher Begriff ist der speziellste Begriff, für den ein Objekt ein Grundbeispiel ist?
realize

Termsubsumtion

19

covers (c,o)

c ist ein Begriff (T-Box),

o ist ein Objekt beschrieben durch $\delta_1 \dots \delta_n$ (A-Box)

covers(c, o)

$\delta_i \in \text{A-Box}$

$\delta_i:(<\text{concept}> o)$

$x = (\text{and } \delta_1 \dots \delta_n)$

subsumes(c, x)

- liefert wahr, dann gilt covers (c,o)
- liefert falsch, dann Fallunterscheidungen

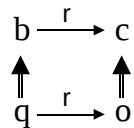
Termsubsumtion

20

Fall 1

Wenn c eine Beschreibung enthält, die in x , der Beschreibung von o fehlt, kann dennoch $\text{covers}(c,o)$ gelten. Nennen wir die fehlenden Beschreibungen in x den Begriff b .

Wenn $(r \sqcap o)$ ist in A-Box und
 $b \sqsubseteq (r \sqcap c)$ ist in T-Box
 dann $\text{covers}(b, q)$ ist wahr und damit $\text{covers}(c,o)$
 sonst falsch



Termsubsumtion

21

Beispiel

T-Box:

Human $\sqsubseteq$ Anything
 Man $\sqsubseteq$ Human
 Woman $\sqsubseteq$ Human
 Set $\sqsubseteq$ Anything
 member $\sqsubseteq$ Anyrelation
 Team $\sqsubseteq$. (and Set (all member Human)
 (atleast 2 member))
 leader $\sqsubseteq$ member
 Modern-Team $\sqsubseteq$. (and Team
 (atmost 4 member)
 (atleast 1 leader)
 (all leader Woman))

A-Box:

(Modern-Team team-a)
 (Man dick)
 (Human mary)
 (member team-a dick)
 (member team-a harry)
 (member team-a mary)
 (leader team-a mary)
 (member team-a (atmost 3))

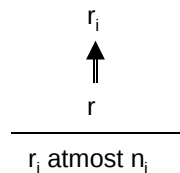
$\text{covers}(\text{Woman} \sqcap \text{mary})?$
 r : leader
 Wenn team-a ein Modern-Team ist,
 ist mary eine Frau.

Termsubsumtion

22

Fall 2

Wenn $\delta_j = (r \sqcap o \text{ (atmost } n))$ fehlt, aber
 $\delta_i = (r_i \sqcap o \text{ (atmost } n_i))$ ist in A-Box
 und $r_i \gg r$ ist in T-Box,
 dann sammle alle $(r_i \sqcap o \text{ (atmost } n_i))$ in der A-Box,
 füge sie x hinzu,
 $\text{subsumes}(c, x')$
 falls es wegen atmost-Verletzung falsch
 $\text{covers}(c,o)$ falsch
 sonst die nächsten Fälle prüfen!



Termsubsumtion

23

Beispiel

o : hans
 x : (Stud hans)
 (beschäftigt_sich hans (atmost 1))
 (beschäftigt_sich hans informatik)
 (Mann hans)
 c :

Termsubsumtion

24

Fall 3

Wenn $(r \text{ o } (\text{atleast } n))$ fehlt, aber
 $(r_i \text{ o } (\text{atleast } n_i))$ in A-Box und
 $r_i \ll r$ in T-Box
dann zähle Werte von r ,
 sammle alle $(r_i \text{ o } (\text{atleast } n_i))$,
 füge sie x hinzu.
 $\text{subsumes}(c, x')$
 falls falsch wegen atleast ist $\text{covers}(c, o)$ falsch
 sonst weiter!

Termsubsumtion

25

Fall 4

Wenn $(r \text{ o } oz_i)$ in A-Box und $(\text{all } r \text{ } c_z)$ T-Box
 nicht klappte,
dann teste, ob alle oz_i genannt sind
 Wenn ja, für alle oz_i $\text{covers}(c_z, oz_i)$ prüfen,
 sonst liefere falsch

Termsubsumtion

26

Realisierungsalgorithmus (realizer)

- 1.) Propagierung
- 2.) Abstraktion
- 3.) Klassifikation

Termsubsumtion

27

Propagierung

- $\delta = (r \text{ o } p)$
- alle $(c_i \text{ o})$ sammeln,
 - deren Wertebereichsbeschränkungen für r und $r_i \gg r$ sammeln,
 - alles als Beschreibung an p reichen
- $\delta = (c \text{ o})$ und
 es gibt $(r_i \text{ o } p)$ in A-Box und c : $(\text{all } r_i \text{ } b_i)$ in T-Box
- sammle alle b_i für r_i und r_j ,
 - $r_j \gg r_i$, als Beschreibung an p reichen.

Termsubsumtion

28

Abstraktion

Die Abstraktion eines Objektes o ist die Konjunktion von:

(and $c_1 \dots c_n$) für alle $(c_i \ o)$ in A-Box

(atleast $n \ r$) Einträge für

$(r_i \ o \dots)$ Einträge und

$(r_i \ o \text{ (atleast } n_i))$ mit $r_i \ll r$

(atmost $n \ r$) für $(r_i \ o \text{ (atmost } n_i))$ Einträge mit $r \ll r_i$

(all $r \ c_z$) falls alle Werte p_i bekannt sind für $(r_i \ o \ p_i)$
mit $r_i \ll r$, Beschreibungen von
 p_i abstrahieren und generalisieren zu c_z .

Termsubsumtion

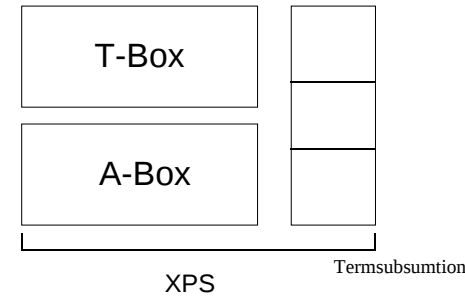
29

Klassifikation

... für alle Abstraktionen aufrufen.

meist: Tiefenbeschränkung der Vorwärtsinferenzen

=> Realisierung ist auch unvollständig

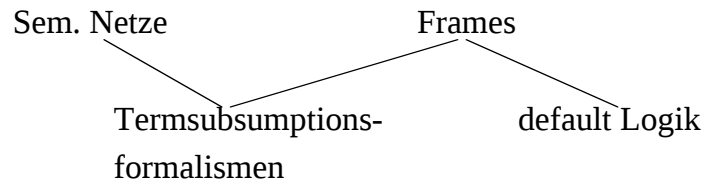


„erklärbare Expertensysteme“

Termsubsumtion

30

Zusammenfassung



T-Box Syntax

Semantik -> ext

subsumes <- Klassifikation

A-Box Syntax

Semantik -> int

covers <- Realisierung

entscheidbar

unvollständig (falls androle) Also: aufpassen bei falsch!

polynomial (falls kein disjoint)

Termsubsumtion

31