Semantic Web Grundlagen

Lösung zur Übung 2: Logik, RDF-Semantik, Datalog

Birte Glimm

WS 2011/2012

Lösung (2.1)**.**

(a) $(p \vee \neg p)$: allgemeingültig

$I(p)$	$I(\neg p)$	$I(p \vee \neg p)$
t	f	t
f	t	t

(b) $((p \vee q) \rightarrow (\neg p \vee \neg q))$: erfülbal & widerlegbar

$I(p)$	$I(q)$	$I(\neg p)$	$I(\neg q)$	$I(p \vee q)$	$I(\neg p \vee \neg q)$	$I((p \vee q) \rightarrow (\neg p \vee \neg q))$
t	t	f	f	t	f	f
t	f	f	t	t	t	t
f	t	t	f	t	t	t
f	f	t	t	f	t	t

(c) $\neg((p \rightarrow q) \leftrightarrow (\neg p \vee q))$: unerüllbar

$I(p)$	$I(q)$	$I(\neg p)$	$I(p \rightarrow q)$	$I(\neg p \vee q)$	$I((p \rightarrow q) \leftrightarrow (\neg p \vee q))$	$I(\neg((p \rightarrow q) \leftrightarrow (\neg p \vee q)))$
t	t	f	t	t	t	f
t	f	f	f	f	t	f
f	t	t	t	t	t	f
f	f	t	t	t	t	f

(d) $(((p \rightarrow q) \rightarrow p) \rightarrow p)$: allgemeingültig

$I(p)$	$I(q)$	$I((p \rightarrow q))$	$I(((p \rightarrow q) \rightarrow p))$	$I((((p \rightarrow q) \rightarrow p) \rightarrow p))$
t	t	t	t	t
t	f	f	t	t
f	t	t	f	t
f	f	t	f	t

(e) $(((p \wedge q) \rightarrow r) \leftrightarrow (p \rightarrow (q \rightarrow r)))$: allgemeingültig

$I(p)$	$I(q)$	$I(r)$	$I((p \wedge q))$	$I(((p \wedge q) \rightarrow r))$
t	t	t	t	t
t	t	f	t	f
t	f	t	f	t
t	f	f	f	t
f	t	t	f	t
f	t	f	f	t
f	f	t	f	t
f	f	f	f	t

$I(q \rightarrow r)$	$I((p \rightarrow (q \rightarrow r)))$	$I((((p \wedge q) \rightarrow r) \leftrightarrow (p \rightarrow (q \rightarrow r))))$
t	t	t
f	f	t
t	t	t
t	t	t
t	t	t
f	t	t
t	t	t
t	t	t

(f) $((p \wedge \neg p) \rightarrow q)$: allgemeingültig

$I(p)$	$I(q)$	$I(\neg p)$	$I((p \wedge \neg p))$	$I(((p \wedge \neg p) \rightarrow q))$
t	t	f	f	t
t	f	f	f	t
f	t	t	f	t
f	f	t	f	t

Lösung (2.2)**.**
Theorie: Eine *Theorie* ist eine Menge von Sätzen.
Logische Konsequenz: α ist logische Konsequenz von $\mathcal{T}$ wenn $\mathcal{T} \models \alpha$.
Äquivalenz: $\alpha \equiv \psi$ wenn $\{\alpha\} \models \psi$ und $\{\psi\} \models \alpha$.

Für beliebige Theorien $\mathcal{T}$ und $\mathcal{S}$ gilt:

(a) Ist eine Formel F allgemeingültig, dann gilt $\mathcal{T} \models F$, d.h. aus jeder Theorie folgen zumindest alle Tautologien.
✓ wahr: Für beliebige Interpretation I, wenn $I \models \mathcal{T}$, dann $I \models F$

(b) Je größer eine logische Theorie ist, desto mehr Modelle hat sie. Das heißt, wenn $\mathcal{T} \subseteq \mathcal{S}$, dann ist jedes Modell von $\mathcal{T}$ auch ein Modell von $\mathcal{S}$.
✖ falsch: Gegenbeispiel: $\mathcal{T} = \{p\}$, $\mathcal{S} = \{p, \neg p\}$. $\mathcal{T}$ ist erüllbar und $\mathcal{S}$ ist unerfüllbar.

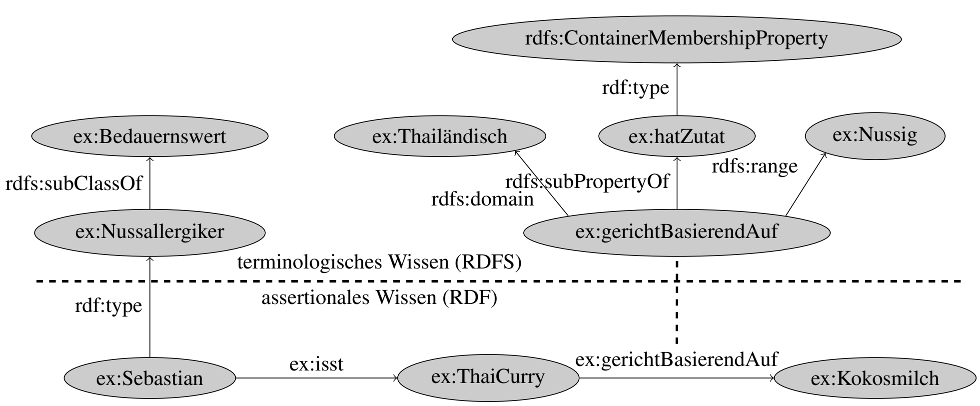
(c) Je größer eine Theorie ist, desto mehr logische Konsequenzen hat sie. Das heißt, wenn $\mathcal{T} \subseteq \mathcal{S}$, dann ist jede logische Konsequenz aus $\mathcal{T}$ auch eine Konsequenz aus $\mathcal{S}$.
✓wahr: "Monotonie der Logik"

(d) Ist $\neg F \in \mathcal{T}$, dann kann $\mathcal{T} \models F$ niemals gelten (wobei F eine beliebige Formel ist).
✖ falsch: im Allgemeinen, denn eine *widersprüchliche*, also *inkonsistente* Theorie hat beliebige Konsequenzen
(✓wahr für *konsistente* Theorien: Für ein beliebiges Modell I einer *konsistenten* Theorie $\mathcal{T}$ und $F' \in \mathcal{T}$ gilt: $I \models F'$; insbesondere, wenn $\neg F \in \mathcal{T}$ und $I \models \mathcal{T}$, dann $I \models \neg F$, also $I \not\models F$ und $\mathcal{T} \models F$ kann niemals gelten.

(e) Sind zwei Theorien unterschiedlich ($\mathcal{T} \neq \mathcal{S}$), dann unterscheiden sie sich auch in wenigstens einer logischen Konsequenz (zum Beispiel, indem es eine Formel F gibt, so dass $\mathcal{T} \models F$ aber $\mathcal{S} \not\models F$).
✖ falsch: Für zwei äquivalente Theorien $\mathcal{T}$ und $\mathcal{S}$ mit $\mathcal{T} \neq \mathcal{S}$, wenn $\mathcal{T} \models F$ (für eine beliebige Formel F), dann ist jedes Modell von $\mathcal{T}$ ein Modell von F. Aber jedes Modoll von $\mathcal{T}$ ist auch ein Modell von $\mathcal{S}$ ($\mathcal{T} \equiv \mathcal{S}$), d.h. $\mathcal{S} \models F$,z.B. $\{p, q\}$ und $\{p \wedge q\}$, $\{p, q\} \neq \{(p \wedge q)\}$ aber $\{p, q\} \models (p \wedge q)$ und $\{(p \wedge q)\} \models (p \wedge q)$.

Lösung (2.3)**.**
Die folgende Interpretation I ist ein Modell des Graphen.

- $\mathrm{IR} = \{a\}$
- $\mathrm{IP} = \{a\}$
- $\mathrm{I_{EXT}}(a) = \{\langle a, a\rangle\}$
- $\mathrm{I_S}(x) = a$, x ist beliebige Resource im Graph.
- $\mathrm{LV} = \mathrm{I_L} = \emptyset$



- I ist eine (einfache) Interpretation: ✓

- I ist eine RDF-interpretation: ✓
- I ist eine RDFS-interpretation: ✓

Lösung (2.4)**.**

- Ein Tripel, welches einfach folgt:

  `ex:ThaiCurry ex:gerichtBasierendAuf ex:Kokosmilch`
  $\Downarrow$ (Regel se1)
  `ex:ThaiCurry ex:gerichtBasierendAuf _:id1`

- Ein Tripel, welches RDF-folgt, aber nicht einfach folgt:

  `ex:ThaiCurry ex:gerichtBasierendAuf ex:Kokosmilch`
  $\Downarrow$ (Regel rdf1)
  `ex:gerichtBasierendAuf rdf:type rdf:Property`

- Ein Tripel, welches RDFS-folgt, aber nicht einfach folgt:

  `ex:ThaiCurry ex:gerichtBasierendAuf ex:Kokosmilch`
  und
  `ex:gerichtBasierendAuf rdfs:domain ex:Thailändisch`
  $\Downarrow$ (Regel rdfs2)
  `ex:ThaiCurry rdf:type ex:Thailändisch`

Lösung (2.5)**.**
Nicht möglich in RDFS.

Lösung (2.6)**.**

(a) `rdfs:Resource rdf:type rdfs:Class .`

$$\frac{\texttt{rdfs:domain rdfs:range rdfs:Class .} \quad \texttt{rdf:type rdfs:domain rdfs:Resource .}}{\texttt{rdfs:Resouce rdf:type rdfs:Class .}}\ \text{rdfs3}$$

(b) `rdfs:Class rdf:type rdfs:Class .`

$$\frac{\texttt{rdfs:range rdfs:range rdfs:Class .} \quad \texttt{rdfs:range rdfs:range rdfs:Class .}}{\texttt{rdfs:Class rdf:type rdfs:Class .}}\ \text{rdfs3}$$

(c) `rdfs:Literal rdf:type rdfs:Class .`

$$\frac{\texttt{rdfs:range rdfs:range rdfs:Class .} \quad \texttt{rdfs:comment rdfs:range rdfs:Literal .}}{\texttt{rdfs:Literal rdf:type rdfs:Class .}}\ \text{rdfs3}$$

(d) rdf:XMLLiteral rdf:type rdfs:Class .

$$\frac{\texttt{rdfs:subClassOf rdfs:domain rdfs:Class .} \quad \texttt{rdf:XMLLiteral rdfs:subClassOf rdfs:Literal .}}{\texttt{rdf:XMLLiteral rdf:type rdfs:Class .}}\ \text{rdfs2}$$

(e) rdfs:Datatype rdf:type rdfs:Class .

$$\frac{\texttt{rdf:type rdfs:range rdfs:Class .} \quad \texttt{rdf:XMLLiteral rdf:type rdfs:Datatype .}}{\texttt{rdfs:Datatype rdf:type rdfs:Class .}}\ \text{rdfs3}$$

(f) rdf:Seq rdf:type rdfs:Class .

$$\frac{\texttt{rdfs:subClassOf rdfs:domain rdfs:Class .} \quad \texttt{rdf:Seq rdfs:subClassOf rdfs:Container .}}{\texttt{rdf:Seq rdf:type rdfs:Class .}}\ \text{rdfs2}$$

(g) rdf:Bag rdf:type rdfs:Class .

$$\frac{\texttt{rdfs:subClassOf rdfs:domain rdfs:Class .} \quad \texttt{rdf:Bag rdfs:subClassOf rdfs:Container .}}{\texttt{rdf:Bag rdf:type rdfs:Class .}}\ \text{rdfs2}$$

(h) rdf:Alt rdf:type rdfs:Class .

$$\frac{\texttt{rdfs:subClassOf rdfs:domain rdfs:Class .} \quad \texttt{rdf:Alt rdfs:subClassOf rdfs:Container .}}{\texttt{rdf:Alt rdf:type rdfs:Class .}}\ \text{rdfs2}$$

(i) rdfs:Container rdf:type rdfs:Class .

$$\frac{\texttt{rdfs:subClassOf rdfs:range rdfs:Class .} \quad \texttt{rdf:Alt rdfs:subClassOf rdfs:Container .}}{\texttt{rdfs:Container rdf:type rdfs:Class .}}\ \text{rdfs3}$$

(j) rdf:List rdf:type rdfs:Class .

$$\frac{\texttt{rdfs:domain rdfs:range rdfs:Class .} \quad \texttt{rdf:first rdfs:domain rdf:List .}}{\texttt{rdf:List rdf:type rdfs:Class .}}\ \text{rdfs3}$$

(k) rdfs:ContainerMembershipProperty rdf:type rdfs:Class .

$$\frac{\texttt{rdfs:subClassOf rdfs:domain rdfs:Class .} \quad \texttt{rdfs:ContainerMembershipProperty rdfs:subClassOf rdfs:Property .}}{\texttt{rdfs:ContainerMembershipProperty rdf:type rdfs:Class .}}\ \text{rdfs2}$$

(l) rdf:Property rdf:type rdfs:Class .

$$\frac{\texttt{rdfs:range rdfs:range rdfs:Class .} \quad \texttt{rdf:subPropertOf rdfs:range rdf:Property .}}{\texttt{rdf:Property rdf:type rdfs:Class .}}\ \text{rdfs3}$$

(m) rdf:Statement rdf:type rdfs:Class .

$$\frac{\texttt{rdfs:domain rdfs:range rdfs:Class .} \quad \texttt{rdf:Subject rdfs:domain rdf:Statement .}}{\texttt{rdf:Statement rdf:type rdfs:Class .}}\ \text{rdfs3}$$

(n) rdfs:domain rdf:type rdf:Property .

$$\frac{\texttt{rdfs:range rdfs:domain rdf:Property .}}{\texttt{rdfs:domain rdf:type rdf:Property .}}\ \text{rdf1}$$

(o) `rdfs:range rdf:type rdf:Property .`

$$\frac{\texttt{rdfs:subPropertyOf rdfs:range rdf:Property .}}{\texttt{rdfs:range rdf:type rdf:Property .}}\ \text{rdfs3}$$

(p) `rdfs:subPropertyOf rdf:type rdf:Property .`

$$\frac{\texttt{rdfs:isDefinedBy rdfs:subPropertyOf rdfs:seeAlso .}}{\texttt{rdfs:subPropertyOf rdf:type rdf:Property .}}\ \text{rdf1}$$

(q) `rdfs:subClassOf rdf:type rdf:Property .`

$$\frac{\texttt{rdf:Alt rdfs:subClassOf rdfs:Container .}}{\texttt{rdfs:subClassOf rdf:type rdf:Property .}}\ \text{rdf1}$$

(r) `rdfs:member rdf:type rdf:Property .`

$$\frac{\texttt{rdfs:range rdfs:domain rdf:Property .}\quad \texttt{rdfs:member rdfs:range rdfs:Resource .}}{\texttt{rdfs:member rdf:type rdf:Property .}}\ \text{rdfs2}$$

(s) `rdfs:seeAlso rdf:type rdf:Property .`

$$\frac{\texttt{rdfs:range rdfs:domain rdf:Property .}\quad \texttt{rdfs:seeAlso rdfs:range rdfs:Resource .}}{\texttt{rdfs:seeAlso rdf:type rdf:Property .}}\ \text{rdfs2}$$

(t) `rdfs:isDefinedBy rdf:type rdf:Property .`

$$\frac{\texttt{rdfs:range rdfs:domain rdf:Property .}\quad \texttt{rdf:isDefinedBy rdfs:range rdfs:Resource .}}{\texttt{rdfs:isDefinedBy rdf:type rdf:Property .}}\ \text{rdfs2}$$

(u) `rdfs:comment rdf:type rdf:Property .`

$$\frac{\texttt{rdfs:range rdfs:domain rdf:Property .}\quad \texttt{rdfs:comment rdfs:range rdfs:Literal .}}{\texttt{rdfs:comment rdf:type rdf:Property .}}\ \text{rdfs2}$$

(v) `rdfs:label rdf:type rdf:Property .`

$$\frac{\texttt{rdfs:range rdfs:domain rdf:Property .}\quad \texttt{rdfs:label rdfs:range rdfs:Literal .}}{\texttt{rdfs:label rdf:type rdf:Property .}}\ \text{rdfs2}$$

Lösung (2.7)**.**

Algorithm 1 D

$d := \emptyset$
repeat
  $d := d \cup \pi_{14}(\sigma_{2=3}(u \times r)) \cup \pi_{14}(\sigma_{2=3}(r \times u)) \cup \pi_{14}(\sigma_{2=3}(d \times d))$
until Fixpunkt erreicht

Iteration 0:	$\emptyset$
Iteration 1:	$\{(00, 11), (11, 22), (22, 33), (00, 22)\}$
Iteration 2:	$\{$Iteration 1$\} \cup \{(00, 22), (11, 33), (00, 33)\}$
Iteration 3:	$\{$Iteration 2$\}$