

Beschreibungslogik | Übung 06

D. Marschner, A. Mahdavi

alma@uni-bremen.de

mail@dennis-marschner.de

Aufgabe 1

1. (20 %) Wendet das konsequenzbasierte Verfahren (Regeln **R1–R4**) für Subsumtion in \mathcal{EL} mit TBoxen an, um folgende Frage zu entscheiden:

Wird A_1 subsumiert von A_2 bzgl. \mathcal{T} , wobei

$$\mathcal{T} = \{A_1 \sqsubseteq \exists r.A_3, A_2 \sqsubseteq A_3, \top \sqsubseteq \exists s.A_2, \exists s.A_3 \sqsubseteq A_1, \exists r.A_1 \sqsubseteq A_2\}$$

<p>R1 $\frac{}{A \sqsubseteq A}$ wenn A in \mathcal{T} vorkommt</p> <p>R2 $\frac{}{A \sqsubseteq \top}$ wenn A in \mathcal{T} vorkommt</p> <p>R3 $\frac{A \sqsubseteq A_1 \quad \dots \quad A \sqsubseteq A_n \quad A_1 \sqcap \dots \sqcap A_n \sqsubseteq B}{A \sqsubseteq B}$</p> <p>R4 $\frac{A \sqsubseteq \exists r.A_1 \quad A_1 \sqsubseteq B_1 \quad \exists r.B_1 \sqsubseteq B}{A \sqsubseteq B}$</p>	<p>Aussprache</p> <p>$\mathcal{T} \models C \sqsubseteq D$</p> <p>D wird von C subsumiert bezüglich der T-Box?</p> <p>$C \sqsubseteq D$</p> <p>C impliziert D</p>
--	--

Wird A_2 subsumiert von A_1 bezüglich TBox \mathcal{T} ?

Frage: $\mathcal{T} \models A_2 \sqsubseteq A_1$? Wobei \mathcal{T} wie folgt definiert ist:

$$\mathcal{T} = \{$$

$$\begin{array}{ll} A_1 \sqsubseteq \exists r. A_3, & (1) \\ A_2 \sqsubseteq A_3, & (2) \\ \top \sqsubseteq \exists s. A_2, & (3) \\ \exists s. A_3 \sqsubseteq A_1, & (4) \\ \exists r. A_1 \sqsubseteq A_2, & (5) \end{array}$$

$$\}$$

Die TBox \mathcal{T} ist bereits in NF.

Algorithmus: Subsumtion mit TBox

- R1** $A_1 \sqsubseteq A_1 \quad A_2 \sqsubseteq A_2 \quad A_3 \sqsubseteq A_3 \quad \top \sqsubseteq \top$ (6)
- R2** $A_1 \sqsubseteq \top \quad A_2 \sqsubseteq \top \quad A_3 \sqsubseteq \top$ (7)
- R3** Zunächst nicht anwendbar auf (1) – (7)
- R4** auf (3) $\top \sqsubseteq \exists s. A_2$, (2) $A_2 \sqsubseteq A_3$, (4) $\exists s. A_3 \sqsubseteq A_1$
 $\top \sqsubseteq A_1$ (8)
- R3** auf (7) $A_2 \sqsubseteq \top$, (8) $\top \sqsubseteq A_1$
 $A_2 \sqsubseteq A_1$ (9)

$$\mathcal{T}^* = \{$$

$$A_1 \sqsubseteq \exists r. A_3,$$

$$\begin{array}{l}
A_2 \sqsubseteq A_3, \\
\top \sqsubseteq \exists s. A_2, \\
\exists s. A_3 \sqsubseteq A_1, \\
\exists r. A_1 \sqsubseteq A_2, \\
A_1 \sqsubseteq A_1, \\
A_2 \sqsubseteq A_2, \\
A_3 \sqsubseteq A_3, \\
\top \sqsubseteq \top, \\
A_1 \sqsubseteq \top, \\
A_2 \sqsubseteq \top, \\
A_3 \sqsubseteq \top, \\
\top \sqsubseteq A_1, \\
A_2 \sqsubseteq A_1 \\
\}
\end{array}$$

Durch Regelanwendung **R4** kam heraus, dass gilt: $\mathcal{T} \models \top \sqsubseteq A_1$.

Durch die anschließende Anwendung von **R3** kam heraus, dass gilt: $\mathcal{T} \models A_2 \sqsubseteq A_1$.

Daher ist die Frage ob $\mathcal{T} \models A_2 \sqsubseteq A_1$ gilt, eindeutig mit "Ja" zu beantworten.

Aufgabe 2

Zusammenfassung:

- R1 & R2 gelöst
- R3 fast vollständig gelöst
- R4 offen
- Aber Vorgehen und Problemstellung genau verstanden

Anmerkung zu unserem Vorgehen/ Teillösung:

Wir haben die Java-OWL API genau analysiert, um zu verstehen, welche Java Funktionen genutzt werden können, um die gegebene Ontology auszulesen und zu manipulieren. Tatsächlich war das Schwierigste die simple aber wichtige Frage, welcher Part von Subclasses jeweils auf der linken und auf der rechten Seite einer Inklusion standen. Leider haben wir dies bis zum Schluss nicht herausgefunden, sodass wir eine Notlösung implementieren mussten. In dieser Lösung nutzen wir die String-Ausgabe eines Subclass Axioms und trennen dieses per REGEX-Abfrage in linken und rechten Teil. Anschließend vergleichen wir es mit den innerNestedClasses dieser Subclass um so herauszufinden, welches Konzept rechts oder links von der Inklusion steht.

Die vergebliche Suche kostete soviel Zeit, dass wir schließlich nicht alle Testfälle sauber implementieren konnten. Dennoch denken wir, die wesentlichen Problemstellungen genau erkannt zu haben.

Für die Regeln R1 und R2 war die Implementierung straight forward.

Für die Regeln R3 und R4 mussten wir mehr Tests durchführen. Schließlich wurde uns klar, dass der sinnvollste Weg ist, Rekursion anzuwenden.

Der entscheidende Lösungsvorgehen ist dabei recht simpel:

Zunächst generieren wir alle möglichen Inklusionen anhand aller existierenden Konzepte und Existenz-Restriktionen. Und dann gehen wir jede mögliche Konstellation durch, indem wir rekursiv den Baum zurücklaufen und fragen, ob die Inklusion gilt. Denn wenn ja, gibt es immer noch eine weitere Inklusion, die ebenfalls den linken Teil der zu prüfenden Inklusion impliziert. Und wenn dieser linke implizierte Teil auch impliziert wird, ja dann ist es logisch, dass transitiv auch die zu prüfende Inklusion gilt!

Nach diesem Schema könnten wir sicher auch R3 und R4 vollständig zuende implementieren, doch leider ging und die Zeit aus.

PS: Hat uns gefreut, endlich mal das Gelernte auch in die Tat umzusetzen, war eine interessante Erfahrung um den Grad zwischen Theorie und Praxis der BL besser einzuschätzen.

Java Log Output

Testfall 0:

```
C:\Users\Dennis\development\DescriptionLogic\blatt6\java_bl_reasoner\tests\test0in.
owl
  OK.  (Ontologie ist nicht in EL-Normalform)
```

Testfall 1:

```
C:\Users\Dennis\development\DescriptionLogic\blatt6\java_bl_reasoner\tests\test1in.
owl
R1 Add:
SubClassOf(<http://www.semanticweb.org/schneider/ontologies/2019/6/test1in#A>
<http://www.semanticweb.org/schneider/ontologies/2019/6/test1in#A>)
R2 Add:
SubClassOf(<http://www.semanticweb.org/schneider/ontologies/2019/6/test1in#A>
owl:Thing)
R1 Add:
SubClassOf(<http://www.semanticweb.org/schneider/ontologies/2019/6/test1in#B>
<http://www.semanticweb.org/schneider/ontologies/2019/6/test1in#B>)
R2 Add:
SubClassOf(<http://www.semanticweb.org/schneider/ontologies/2019/6/test1in#B>
owl:Thing)
  OK.
```

Testfall 2:

```
C:\Users\Dennis\development\DescriptionLogic\blatt6\java_bl_reasoner\tests\test2in.
owl
R1 Add: SubClassOf(owl:Thing owl:Thing)
R2 Add: SubClassOf(owl:Thing owl:Thing)
R1 Add:
SubClassOf(<http://www.semanticweb.org/schneider/ontologies/2019/6/test2in#A>
<http://www.semanticweb.org/schneider/ontologies/2019/6/test2in#A>)
R2 Add:
SubClassOf(<http://www.semanticweb.org/schneider/ontologies/2019/6/test2in#A>
owl:Thing)
R1 Add:
SubClassOf(<http://www.semanticweb.org/schneider/ontologies/2019/6/test2in#B>
<http://www.semanticweb.org/schneider/ontologies/2019/6/test2in#B>)
R2 Add:
SubClassOf(<http://www.semanticweb.org/schneider/ontologies/2019/6/test2in#B>
owl:Thing)
R3 Add:
SubClassOf(<http://www.semanticweb.org/schneider/ontologies/2019/6/test2in#A>
owl:Thing)
R3 Add:
SubClassOf(<http://www.semanticweb.org/schneider/ontologies/2019/6/test2in#B>
owl:Thing)
  OK.
```

Testfall 3:

```
C:\Users\Dennis\development\DescriptionLogic\blatt6\java_bl_reasoner\tests\test3in.
owl
```

```

R1 Add:
SubClassOf(<http://www.semanticweb.org/schneider/ontologies/2019/6/test3in#A>
<http://www.semanticweb.org/schneider/ontologies/2019/6/test3in#A>)
R2 Add:
SubClassOf(<http://www.semanticweb.org/schneider/ontologies/2019/6/test3in#A>
owl:Thing)
R1 Add:
SubClassOf(<http://www.semanticweb.org/schneider/ontologies/2019/6/test3in#A1>
<http://www.semanticweb.org/schneider/ontologies/2019/6/test3in#A1>)
R2 Add:
SubClassOf(<http://www.semanticweb.org/schneider/ontologies/2019/6/test3in#A1>
owl:Thing)
R1 Add:
SubClassOf(<http://www.semanticweb.org/schneider/ontologies/2019/6/test3in#B>
<http://www.semanticweb.org/schneider/ontologies/2019/6/test3in#B>)
R2 Add:
SubClassOf(<http://www.semanticweb.org/schneider/ontologies/2019/6/test3in#B>
owl:Thing)
R3 Add:
SubClassOf(<http://www.semanticweb.org/schneider/ontologies/2019/6/test3in#A>
<http://www.semanticweb.org/schneider/ontologies/2019/6/test3in#A1>)
R3 Add:
SubClassOf(<http://www.semanticweb.org/schneider/ontologies/2019/6/test3in#A>
<http://www.semanticweb.org/schneider/ontologies/2019/6/test3in#B>)
R3 Add:
SubClassOf(<http://www.semanticweb.org/schneider/ontologies/2019/6/test3in#A1>
<http://www.semanticweb.org/schneider/ontologies/2019/6/test3in#B>)
OK.

```

Testfall 4:

C:\Users\Dennis\development\DescriptionLogic\blatt6\java_bl_reasoner\tests\test4in.
owl

```

R1 Add:
SubClassOf(<http://www.semanticweb.org/schneider/ontologies/2019/6/test4in#A1>
<http://www.semanticweb.org/schneider/ontologies/2019/6/test4in#A1>)
R2 Add:
SubClassOf(<http://www.semanticweb.org/schneider/ontologies/2019/6/test4in#A1>
owl:Thing)
R1 Add:
SubClassOf(<http://www.semanticweb.org/schneider/ontologies/2019/6/test4in#A3>
<http://www.semanticweb.org/schneider/ontologies/2019/6/test4in#A3>)
R2 Add:
SubClassOf(<http://www.semanticweb.org/schneider/ontologies/2019/6/test4in#A3>
owl:Thing)
R1 Add:
SubClassOf(<http://www.semanticweb.org/schneider/ontologies/2019/6/test4in#B>
<http://www.semanticweb.org/schneider/ontologies/2019/6/test4in#B>)
R2 Add:
SubClassOf(<http://www.semanticweb.org/schneider/ontologies/2019/6/test4in#B>
owl:Thing)
R1 Add:
SubClassOf(<http://www.semanticweb.org/schneider/ontologies/2019/6/test4in#A>
<http://www.semanticweb.org/schneider/ontologies/2019/6/test4in#A>)
R2 Add:
SubClassOf(<http://www.semanticweb.org/schneider/ontologies/2019/6/test4in#A>
owl:Thing)
R1 Add:
SubClassOf(<http://www.semanticweb.org/schneider/ontologies/2019/6/test4in#A2>
<http://www.semanticweb.org/schneider/ontologies/2019/6/test4in#A2>)
R2 Add:
SubClassOf(<http://www.semanticweb.org/schneider/ontologies/2019/6/test4in#A2>
owl:Thing)
R3 Add:
SubClassOf(<http://www.semanticweb.org/schneider/ontologies/2019/6/test4in#A>
<http://www.semanticweb.org/schneider/ontologies/2019/6/test4in#A1>)
R3 Add:
SubClassOf(<http://www.semanticweb.org/schneider/ontologies/2019/6/test4in#A>
<http://www.semanticweb.org/schneider/ontologies/2019/6/test4in#A3>)

```

R3 Add:
SubClassOf(<http://www.semanticweb.org/schneider/ontologies/2019/6/test4in#A>
<http://www.semanticweb.org/schneider/ontologies/2019/6/test4in#A2>)

FEHLER:

Folgende Subsumtion wird vom Reasoner fälschlicherweise nicht erzeugt:

SubClassOf(<A>)

Testfall 5:

C:\Users\Dennis\development\DescriptionLogic\blatt6\java_bl_reasoner\tests\test5in.
owl

R1 Add:

SubClassOf(<http://www.semanticweb.org/schneider/ontologies/2019/6/test5in#B>
<http://www.semanticweb.org/schneider/ontologies/2019/6/test5in#B>)

R2 Add:

SubClassOf(<http://www.semanticweb.org/schneider/ontologies/2019/6/test5in#B>
owl:Thing)

R1 Add:

SubClassOf(<http://www.semanticweb.org/schneider/ontologies/2019/6/test5in#A1>
<http://www.semanticweb.org/schneider/ontologies/2019/6/test5in#A1>)

R2 Add:

SubClassOf(<http://www.semanticweb.org/schneider/ontologies/2019/6/test5in#A1>
owl:Thing)

R1 Add:

SubClassOf(<http://www.semanticweb.org/schneider/ontologies/2019/6/test5in#A>
<http://www.semanticweb.org/schneider/ontologies/2019/6/test5in#A>)

R2 Add:

SubClassOf(<http://www.semanticweb.org/schneider/ontologies/2019/6/test5in#A>
owl:Thing)

FEHLER:

Folgende Subsumtion wird vom Reasoner fälschlicherweise nicht erzeugt:

SubClassOf(<A>)

Testfall 6:

C:\Users\Dennis\development\DescriptionLogic\blatt6\java_bl_reasoner\tests\test6in.
owl

R1 Add:

SubClassOf(<http://www.semanticweb.org/schneider/ontologies/2019/6/test6in#A>
<http://www.semanticweb.org/schneider/ontologies/2019/6/test6in#A>)

R2 Add:

SubClassOf(<http://www.semanticweb.org/schneider/ontologies/2019/6/test6in#A>
owl:Thing)

R1 Add:

SubClassOf(<http://www.semanticweb.org/schneider/ontologies/2019/6/test6in#B>
<http://www.semanticweb.org/schneider/ontologies/2019/6/test6in#B>)

R2 Add:

SubClassOf(<http://www.semanticweb.org/schneider/ontologies/2019/6/test6in#B>
owl:Thing)

R1 Add:

SubClassOf(<http://www.semanticweb.org/schneider/ontologies/2019/6/test6in#B1>
<http://www.semanticweb.org/schneider/ontologies/2019/6/test6in#B1>)

R2 Add:

SubClassOf(<http://www.semanticweb.org/schneider/ontologies/2019/6/test6in#B1>
owl:Thing)

R1 Add:

SubClassOf(<http://www.semanticweb.org/schneider/ontologies/2019/6/test6in#A1>
<http://www.semanticweb.org/schneider/ontologies/2019/6/test6in#A1>)

R2 Add:

SubClassOf(<http://www.semanticweb.org/schneider/ontologies/2019/6/test6in#A1>
owl:Thing)

R3 Add:

SubClassOf(<http://www.semanticweb.org/schneider/ontologies/2019/6/test6in#A1>
<http://www.semanticweb.org/schneider/ontologies/2019/6/test6in#B1>)

FEHLER:

Folgende Subsumtion wird vom Reasoner fälschlicherweise nicht erzeugt:

SubClassOf(<A>)

Testfall 7:

```
C:\Users\Dennis\development\DescriptionLogic\blatt6\java_bl_reasoner\tests\test7in.
owl
R1 Add:
SubClassOf(<http://www.semanticweb.org/schneider/ontologies/2019/7/test7in#A1>
<http://www.semanticweb.org/schneider/ontologies/2019/7/test7in#A1>)
R2 Add:
SubClassOf(<http://www.semanticweb.org/schneider/ontologies/2019/7/test7in#A1>
owl:Thing)
R1 Add:
SubClassOf(<http://www.semanticweb.org/schneider/ontologies/2019/7/test7in#A2>
<http://www.semanticweb.org/schneider/ontologies/2019/7/test7in#A2>)
R2 Add:
SubClassOf(<http://www.semanticweb.org/schneider/ontologies/2019/7/test7in#A2>
owl:Thing)
R1 Add:
SubClassOf(<http://www.semanticweb.org/schneider/ontologies/2019/7/test7in#A3>
<http://www.semanticweb.org/schneider/ontologies/2019/7/test7in#A3>)
R2 Add:
SubClassOf(<http://www.semanticweb.org/schneider/ontologies/2019/7/test7in#A3>
owl:Thing)
R3 Add:
SubClassOf(<http://www.semanticweb.org/schneider/ontologies/2019/7/test7in#A2>
<http://www.semanticweb.org/schneider/ontologies/2019/7/test7in#A1>)
FEHLER:
Folgende Subsumtionen werden vom Reasoner fälschlicherweise nicht erzeugt:
    SubClassOf(<A1> <A2>)
    SubClassOf(<A2> <A3>)
    SubClassOf(<A1> <A3>)

Testfall 8:

C:\Users\Dennis\development\DescriptionLogic\blatt6\java_bl_reasoner\tests\test8in.
owl
FEHLER: Testontologie
C:\Users\Dennis\development\DescriptionLogic\blatt6\java_bl_reasoner\tests\test8out
.owl fehlt!

Process finished with exit code 0
```