Beschreibungslogik

Übungsblatt 3

Abgabe bis 21. 5. 2017, 23:59 Uhr in Stud.IP, Ordner „Abgabe Übungsblatt 3“, im PDF-Format. Bitte nur eine Abgabe pro Gruppe, Lizenz „Selbst verfasstes, nicht publiziertes Werk“.

---

1. (20 %) Verwende den Tableau-Algorithmus für $\mathcal{ALC}$ aus der Vorlesung, um die Erfüllbarkeit der folgenden Konzepte zu entscheiden.

a) $C_0 = \exists r.A \sqcap \exists r.B \sqcap \forall r.\exists r.A \sqcap \forall r.\forall r.\neg B$

b) $C_0 = \neg(\forall r.\neg A \sqcup \exists r.B) \sqcap \forall r.\neg(A \sqcap \neg B)$

Gib an, welche Regeln in welcher Reihenfolge worauf angewendet werden. Gib im Falle von Erfüllbarkeit außerdem ein Modell gemäß dem Beweis von Theorem 4.8 an.

2. (20 %) Du hast einen vollen Semesterplan und es gibt viel zu erledigen. Alle Aufgaben sind in Deinem Organizer gespeichert, zusammen mit einer (ganzzahligen) Priorität zwischen 1 (niedrig) und 100 (hoch). Du arbeitest eine Aufgabe nach der anderen ab. Wenn Du eine Aufgabe abgearbeitet hast, kann es sein, dass eine Reihe von Teilaufgaben zu bearbeiten sind. In diesem Fall löschst Du nicht nur die momentane Aufgabe aus dem Organizer, sondern ersetzt sie durch die Teilaufgaben. Jede davon hat eine strikt niedrigere Priorität als die ursprüngliche Aufgabe.

Zeige mittels einer Multimengenordnung, dass Du schließlich alle Aufgaben erledigen wirst.

3. (20 %) Betrachte die folgende Erweiterung des Tableau-Algorithmus aus der Vorlesung (ohne TBoxen) auf die Beschreibungslogik $\mathcal{ALCQ}$:

1. zusätzliche $\geq$-Regel:
   - Wähle $v \in V$ und $\geqslant n\, r.C \in \mathcal{L}(v)$, so dass es $< n$ Knoten $v' \in V$ gibt mit $(v, r, v') \in E$ und $C \in \mathcal{L}(v')$.
   - Erweitere V um Knoten $v_1, \ldots, v_n$ und E um $(v, r, v_1), \ldots, (v, r, v_n)$, und setze $\mathcal{L}(v_i) = \{C\}$ für $1 \leq i \leq n$.
2. zusätzliche Art von „offensichtlichem Widerspruch“:
   - Es gibt Knoten $v, v_1, \ldots, v_{n+1}$, so dass $\leqslant n\, r.C \in \mathcal{L}(v)$, $(v, r, v_i) \in E$ und $C \in \mathcal{L}(v_i)$ für $1 \leq i \leq n+1$.

Zeige, dass der skizzierte Algorithmus auf den folgenden Eingabekonzepten ein falsches Ergebnis liefert:

a) $C_0 = (\geqslant 2\, r.(A \sqcap B)) \sqcap (\geqslant 2\, r.(A \sqcap B')) \sqcap (\leqslant 3\, r.A)$

b) $C_0 = (\geqslant 3\, r.A) \sqcap (\leqslant 1\, r.B) \sqcap (\leqslant 1\, r.\neg B)$

Bitte wenden.

4. (20 %) Verwende den Tableau-Algorithmus für $\mathcal{ALC}$ mit TBoxen aus der Vorlesung um zu entscheiden, ob

a) $C_0 = A$ erfüllbar bzgl. $\mathcal{T}_1 = \{\top \sqsubseteq \exists r.\exists r.A \sqcap \forall r.A' \sqcap (\neg A \sqcup \neg A')\}$ ist;

b) $C_0 = A \sqcap B' \sqcap \forall r.(B \sqcap \forall r.B')$ erfüllbar bzgl. $\mathcal{T}_2 = \{\top \sqsubseteq \exists r.A \sqcap \exists s.A\}$ ist;

c) die Subsumtion $\mathcal{T}_3 \models \top \sqsubseteq \mathsf{HappyStudent}$ gilt, wobei $\mathcal{T}_3 = \{\mathsf{HappyStudent} \equiv \forall \mathsf{takesCourse}.\mathsf{Fun},\ \exists \mathsf{takesCourse}.\top \sqsubseteq \mathsf{HappyStudent}\}$.

Im Falle von Erfüllbarkeit bzw. Nicht-Subsumtion gib das Modell an, das sich aus der Konstruktion im Beweis von Theorem 4.13 ergibt.

5. (20 %) Benutze Protégé 5 und einen Reasoner (z. B. `Hermit`, `FaCT++` oder `Pellet`), um Deine Ergebnisse aus Aufgabe 4 zu überprüfen. Schreibe dazu jeweils eine OWL-Ontologie mit den Axiomen aus $\mathcal{T}_i$ verfahre wie folgt.

a) Klassifiziere $\mathcal{T}_1$ und teste, ob A in der „Class hierarchy (inferred)" rot eingefärbt ist und unter `owl:Nothing` erscheint (genau dann ist A *un*erfüllbar bzgl. $\mathcal{T}_1$).

b) Füge die Definition $C_0 \equiv A \sqcap B' \sqcap \forall r.(B \sqcap \forall r.B')$ zu $\mathcal{T}_2$ hinzu und teste dann wie unter a) die Erfüllbarkeit von C_0.

c) Füge die Definition $D \equiv$ `HappyStudent` zu $\mathcal{T}_3$ hinzu und teste dann, ob nach dem Klassifizieren in der „Class hierarchy (inferred)" D neben `owl:Thing` erscheint.

Lege Deiner Abgabe die drei Ontologien und einen Screenshot der „Class hierarchy (inferred)" nach dem Klassifizieren bei.

Benutze bei Bedarf die in Aufgabe 5 b) auf Blatt 1 genannten Hilfsmittel.

6. Zusatzaufgabe (20 %) Mit $\mathcal{ALC}_{\mathsf{trans}}$ bezeichnen wir die Erweiterung von $\mathcal{ALC}$ um transitive Rollen, d. h. die TBox darf nun zusätzlich zu Konzeptinklusionen auch Zusicherungen der Form $\mathsf{trans}(r)$ enthalten, wobei r ein Rollenname ist. Eine Interpretation $\mathcal{I}$ erfüllt $\mathsf{trans}(r)$, wenn $r^{\mathcal{I}}$ eine transitive Relation ist.

Sei $\mathcal{T}$ eine $\mathcal{ALC}_{\mathsf{trans}}$-TBox der Form $\{\top \sqsubseteq C_{\mathcal{T}}\}$ mit $C_{\mathcal{T}}$ in NNF. Wir definieren eine $\mathcal{ALC}$ TBox $\mathcal{T}^*$ wie folgt:

- $\mathcal{T}^*$ enthält $\top \sqsubseteq C_{\mathcal{T}}$.
- Für jedes $\forall r.C \in \mathrm{sub}(C_{\mathcal{T}})$ mit $\mathsf{trans}(r) \in \mathcal{T}$ enthält $\mathcal{T}^*$ die Konzeptinklusion $\forall r.C \sqsubseteq \forall r.\forall r.C$.

a) Beweise, dass für alle Konzeptnamen A gilt:

$$A \text{ ist erfüllbar bzgl. } \mathcal{T} \quad \text{gdw.} \quad A \text{ ist erfüllbar bzgl. } \mathcal{T}^*$$

b) Wie kann man diese Reduktion nutzen, um mit Hilfe des Tableau-Algorithmus für $\mathcal{ALC}$ Erfüllbarkeit in $\mathcal{ALC}_{\mathsf{trans}}$ (mit TBoxen) zu entscheiden? Erkläre kurz.