Assignment#5

Yizheng Zhao

June 13, 2021

Note: Assignment#5, due on 14:00 July 10, contributes to 10% of the total mark of the course.

Q1. Consider the database instance $\mathcal{D}_{\mathsf{PGBBW}}$ (PGBBW stands for Pleasant Goat and Big Big Wolf) given by:

Sheep(weslie)   Sheep(slowy)   LazySheep(paddi)

Sheep(tibbie)   BrownSheep(fitty)   Sheep(jonie)

Wolf(wolffy)   Wolf(wolnie)   Wolf(wilie)

hasFriend(weslie, slowy)   hasFriend(tibbie, jonie)

hasEnemy(paddi, wolffy)   hasWife(wolffy, wolnie)   hasSon(wolnie, wilie)

We query $\mathcal{D}_{\mathsf{PGBBW}}$ under closed world assumption (relational database semantics) and under open world assumption. Recall that under the closed world assumption we consider the interpretation $\mathcal{I} := \mathcal{I}_{\mathcal{D}_{\mathsf{PGBBW}}}$ defined as follows:

- $\Delta^{\mathcal{I}} = \{$weslie, slowy, paddi, tibbie, fitty, jonie, wolffy, wolnie, wilie$\}$
- Sheep$^{\mathcal{I}} = \{$weslie, slowy, tibbie, jonie$\}$
- LazySheep$^{\mathcal{I}} = \{$paddi$\}$
- BrownSheep$^{\mathcal{I}} = \{$fitty$\}$
- Wolf$^{\mathcal{I}} = \{$wolffy, wolnie, wilie$\}$
- hasFriend$^{\mathcal{I}} = \{($weslie, slowy$), ($tibbie, jonie$)\}$
- hasEnemy$^{\mathcal{I}} = \{($paddi, wolffy$)\}$
- hasWife$^{\mathcal{I}} = \{($wolffy, wolnie$)\}$
- hasSon$^{\mathcal{I}} = \{($wolnie, wilie$)\}$

Consider the following Boolean queries (in description logic notation).

(a) Sheep(fitty)

(b) Sheep(wolnie)

(c) Sheep(paddi)

(d) $\neg$Sheep(paddi)

(e) ($\exists$hasFriend.$\top$)(weslie)

(f) ($\exists$hasFriend.Sheep)(weslie)

(g) ($\exists$hasFriend.LazySheep)(weslie)

(h) (BrownSheep $\sqcap$ $\neg$LazySheep)(fitty)

(i) (BrownSheep $\sqcap$ $\neg$Sheep)(fitty)

(j) Sheep(wilie)

(k) ($\exists$hasSon.$\neg$Sheep)(wolnie)

(l) ($\exists$hasEnemy.$\exists$hasWife.Wolf)(paddi)

- Write those Boolean queries in first-order logic (FOL) notation. (Note that for many queries there is no difference between description logic notation and FOL notation).
  (a) Sheep(fitty)
  (b) Sheep(wolnie)
  (c) Sheep(paddi)
  (d) $\neg$Sheep(paddi)
  (e) $\exists y(hasFriend(weslie, y))$
  (f) $\exists y(hasFriend(weslie, y) \wedge Sheep(y))$
  (g) $\exists y(hasFriend(weslie, y) \wedge LazySheep(y))$
  (h) BrownSheep(fitty)$\wedge\neg$LazySheep(fitty)
  (i) BrownSheep(fitty)$\wedge\neg$Sheep(fitty)
  (j) Sheep(wilie)
  (k) $\exists y(hasSon(wolnie, y) \wedge (\neg Sheep(y)))$
  (l) $\exists z(hasEnemy(paddi, z) \wedge (\exists y(hasWife(z, y) \wedge Wolf(y))))$

- Query answering under closed world assumption: check for each Boolean F whether the answer to the query F given by $\mathcal{D}_{\mathsf{PGBBW}}$ is "Yes" or "No". In other words, check whether $\mathcal{I} \models F$ or $\mathcal{I} \models \neg F$.
  (a) No
  (b) No
  (c) No
  (d) Yes
  (e) Yes
  (f) Yes

(g) No
(h) Yes
(i) Yes
(j) No
(k) Yes
(l) Yes

- Query answering under open world assumption: check for each Boolean query F whether the certain answer to F given by $\mathcal{D}_{\mathsf{PGBBW}}$ is "Yes", "No", or "Don't know". In other words, check whether $\mathcal{D} \models F$ or $\mathcal{D} \models \neg F$ or neither of these hold.
(a) Don't know
(b) Don't know
(c) Don't know
(d) Don't know
(e) Yes
(f) Yes
(g) Don't know
(h) Don't know
(i) Don't know
(j) Don't know
(k) Don't know
(l) Yes

Consider the following non-Boolean queries F_i:

(a) $F_1(x) = \mathsf{Wolf}(x)$

(b) $F_2(x) = \neg\mathsf{Sheep}(x)$

(c) $F_3(x, y) = \mathsf{hasFriend}(x, y)$

(d) $F_4(x) = \mathsf{Sheep}(x) \wedge \neg\mathsf{hasFriend}(x, \mathsf{jonie})$

For each query F_i, give

- for closed world assumption: answer(F_i,$\mathcal{D}_{\mathsf{PGBBW}}$);
- for open world assumption: certanswer(F_i,$\mathcal{D}_{\mathsf{PGBBW}}$).

  (a) answer(F_1,$\mathcal{D}_{\mathsf{PGBBW}}$)={wolffy,wolnie,wilie}
  certanswer(F_1,$\mathcal{D}_{\mathsf{PGBBW}}$)={wolffy, wolnie, wilie}

  (b) answer(F_2,$\mathcal{D}_{\mathsf{PGBBW}}$)={paddi,fitty,wolffy,wolnie,wilie}
  certanswer(F_2,$\mathcal{D}_{\mathsf{PGBBW}}$)=$\emptyset$

(c) answer(F_3,$\mathcal{D}_{\mathsf{PGBBW}}$)={(weslie,slowy),(tibbie,jonie)}
certanswer(F_3,$\mathcal{D}_{\mathsf{PGBBW}}$)={(weslie,slowy),(tibbie,jonie)}

(d) answer(F_4,$\mathcal{D}_{\mathsf{PGBBW}}$)={weslie,slowy,jonie}
certanswer(F_4,$\mathcal{D}_{\mathsf{PGBBW}}$)=$\emptyset$

Q2. Following Q1, consider now the TBox $\mathcal{T}$ given as:

$$
\begin{aligned}
\mathsf{LazySheep} &\sqsubseteq \mathsf{Sheep}\\
\mathsf{LazySheep} \sqcap \mathsf{BrownSheep} &\sqsubseteq \bot\\
\mathsf{Sheep} \sqcap \mathsf{Wolf} &\sqsubseteq \bot\\
\top &\sqsubseteq \forall \mathsf{hasFriend}.\mathsf{Sheep}\\
\exists \mathsf{hasFriend}.\mathsf{Sheep} &\sqsubseteq \mathsf{Sheep}\\
\mathsf{Sheep} &\sqsubseteq \exists \mathsf{hasFriend}.\top
\end{aligned}
$$

Fill out the table below with the answers "Yes", "No" or "Don't know" to the Boolean queries.

Query	Answer for $\mathcal{I}$	Certain Answer for $\mathcal{D}_{\mathsf{PGBBW}}$	Certain Answer for $(\mathcal{T}, \mathcal{D}_{\mathsf{PGBBW}})$
LazySheep(paddi)	Yes	Yes	Yes
LazySheep(fitty)	No	Don't know	No
Sheep(paddi)	No	Don't know	Yes
¬Sheep(paddi)	Yes	Don't know	No
BrownSheep(willie)	No	Don't know	Don't know
Wolf(fitty)	No	Don't know	Don't know
∃hasFriend.⊤(fitty)	No	Don't know	Don't know
∀hasFriend.⊤(wolffy)	Yes	Yes	Yes
∀hasFriend.∃hasFriend.⊤(wolnie)	Yes	Don't know	Yes
∃hasFriend.∀hasFriend.⊤(jonie)	No	Don't know	Yes

Q3. Consider the $\mathcal{EL}$ TBox $\mathcal{T}$:

$$
\begin{aligned}
\mathsf{FootballPlayer} &\sqsubseteq \exists \mathsf{plays\_for}.\mathsf{Team}\\
\mathsf{BasketballPlayer} &\sqsubseteq \exists \mathsf{plays\_for}.\mathsf{Team}\\
\mathsf{VolleyballPlayer} &\sqsubseteq \exists \mathsf{plays\_for}.\mathsf{Team}\\
\mathsf{Team} &\sqsubseteq \exists \mathsf{managed\_by}.\mathsf{Manager}\\
\mathsf{Manager} &\sqsubseteq \mathsf{Employee}\\
\mathsf{Manager} &\sqsubseteq \exists \mathsf{managed\_by}.\mathsf{Manager}
\end{aligned}
$$

and the ABox $\mathcal{A}$:

$$
\begin{array}{rl}
\mathsf{FootballPlayer(ronaldo)} & \mathsf{BasketballPlayer(jordan)}\\
\mathsf{VolleyballPlayer(zhuting)} & \mathsf{Team(china)}\\
\mathsf{managed\_by(china, langping)} &
\end{array}
$$

(1) Compute the interpretation $\mathcal{I}_{\mathcal{T},\mathcal{A}}$ as described in the lecture slides.

$\Delta^{\mathcal{I}_{\mathcal{T},\mathcal{A}}} = \{ronaldo, jordan, zhuting, china, langping,$
$d_{FootballPlayer}, d_{BasketballPlayer}, d_{VolleyballPlayer}, d_{Team}, d_{Manager}, d_{Employee}\}$

Initialize S and R:

$S(ronaldo) = \{FootballPlayer\}$
$S(jordan) = \{BasketballPlayer\}$,
$S(zhuting) = \{VolleyballPlayer\}$
$S(china) = \{Team\}$
$S(langping) = \emptyset$
$R(managed\_by) = \{(china, langping)\}$
$R(plays\_for) = \emptyset$
$S(d_{FootballPlayer}) = \{FootballPlayer\}$
$S(d_{BasketballPlayer}) = \{BasketballPlayer\}$
$S(d_{VolleyballPlayer}) = \{VolleyballPlayer\}$
$S(d_{Team}) = \{Team\}$
$S(d_{Manager}) = \{Manager\}$
$S(d_{Employee}) = \{Employee\}$

index the axioms 1-6
for axiom 5,we use simple R:
$S(d_{Manager}) = S(d_{Manager}) \cup \{Employee\} = \{Manager, Employee\}$

axioms 1,2,3,4,6 are the same form as $A \sqsubseteq \exists r.B$,which means the only rule we can use is rightR for R(r)
now since all S have been desided,we have:
$S(d_{FootballPlayer}) = \{FootballPlayer, ronaldo\}, S(d_{BasketballPlayer}) = \{BasketballPlayer, jordan\}$
$S(d_{VolleyballPlayer}) = \{VolleyballPlayer, zhuting\}, S(d_{Team}) = \{Team, china\}$
$S(d_{Manager}) = \{Manager, Employee\}, S(d_{Employee}) = \{Employee\}$

using rightR rule to the remaining axioms, we have:
$R(plays\_for) = \{(d_{FootballPlayer}, d_{Team}), (d_{BasketballPlayer}, d_{Team}), (d_{VolleyballPlayer}, d_{Team})$
$(ronaldo, d_{Team}), (jordan, d_{Team}), (zhuting, d_{Team}))\}$
$R(managed\_by) = \{(china, langping), (d_{Manager}, d_{Manager}), (china, d_{Manager}), (d_{Team}, d_{Manager})\}$

finally we have $\mathcal{I}^{\mathcal{T},\mathcal{A}}$ as follows:
$FootballPlayer^{\mathcal{I}} = \{d_{FootballPlayer}, ronaldo\}$
$BasketballPlayer^{\mathcal{I}} = \{d_{BasketballPlayer}, jordan\}$
$VolleyballPlayer^{\mathcal{I}} = \{d_{VolleyballPlayer}, zhuting\}$
$Team^{\mathcal{I}} = \{d_{Team}, china\}$
$Manager^{\mathcal{I}} = \{d_{Manager}, d_{Employee}\}$
$Employee^{\mathcal{I}} = \{d_{Employee}\}$
$plays\_for^{\mathcal{I}} = \{(d_{FootballPlayer}, d_{Team}), (d_{BasketballPlayer}, d_{Team}), (d_{VolleyballPlayer}, d_{Team})$
$(ronaldo, d_{Team}), (jordan, d_{Team}), (zhuting, d_{Team})\}$
$managed\_by^{\mathcal{I}} = \{(china, langping), (d_{Manager}, d_{Manager}), (china, d_{Manager}), (d_{Team}, d_{Manager})\}$

(2) For $\mathcal{EL}$ concept queries, we know that $\mathcal{I}_{\mathcal{T},\mathcal{A}}$ gives the answer "Yes" iff $(\mathcal{T},\mathcal{A})$ gives the certain answer "Yes". Check this for the queries:

- $\exists$plays_for.Team(zhuting);
- $\exists$managed_by.Manager(zhuting);
- $\exists$plays_for.$\exists$managed_by.Manager(zhuting).

$\exists plays\_for.Team(zhuting)$: both of them give Yes
$\exists managed\_by.Manager(zhuting)$: $\mathcal{I}_{\mathcal{T},\mathcal{A}}$: No; $(\mathcal{T},\mathcal{A})$: Don't know
$\exists plays\_for.\exists managed\_by.Manager(zhuting)$: both of them give Yes

(3) For more complex queries, $\mathcal{I}_{\mathcal{T},\mathcal{A}}$ can give the answer "Yes" even if $(\mathcal{T},\mathcal{A})$ does not give the certain answer "Yes". Check this for:

- $F(x,y) = \exists z.(\text{plays\_for}(x,z) \wedge \text{plays\_for}(y,z))$.
- $F = \exists x.\text{managed\_by}(x,x)$.

Answer:

- $F(x,y)$: $(\mathcal{T},\mathcal{A}) \not\models F(x,y)$, but both jordan and zhuting play for d_{Team} in $\mathcal{I}_{\mathcal{T},\mathcal{A}}$,so $\mathcal{I}_{\mathcal{T},\mathcal{A}} \models F(jordan, zhuting)$
- F: $(\mathcal{T},\mathcal{A}) \not\models F$, but $(d_{Manager}, d_{Manager})$ is in $\mathcal{I}_{\mathcal{T},\mathcal{A}}$, so $\mathcal{I}_{\mathcal{T},\mathcal{A}} \models F$

Q4. Let $\mathcal{I}$ be an interpretation and Σ a signature. The Σ-*reduct* $\mathcal{I}|_\Sigma$ of $\mathcal{I}$ is the interpretation obtained from $\mathcal{I}$ by setting:

- $\Delta^{\mathcal{I}|_\Sigma} := \Delta^{\mathcal{I}}$
- $X^{\mathcal{I}|_\Sigma} := X^{\mathcal{I}}$, for all $X \in \Sigma$;
- $X^{\mathcal{I}|_\Sigma}$ is undefined for all $X \notin \Sigma$.

Two interpretations $\mathcal{I}$ and $\mathcal{J}$ *coincide* on a signature Σ if $\mathcal{I}|_\Sigma = \mathcal{J}|_\Sigma$.

Definition 1 (Σ-inseparability)**.** *Let $\mathcal{T}_1$ and $\mathcal{T}_2$ be two TBoxes and Σ a signature. We say that $\mathcal{T}_1$ and $\mathcal{T}_2$ are* Σ-inseparable*, write $\mathcal{T}_1 \equiv_\Sigma \mathcal{T}_2$, if $\{\mathcal{I}|_\Sigma \mid \mathcal{I} \models \mathcal{T}_1\} = \{\mathcal{I}|_\Sigma \mid \mathcal{I} \models \mathcal{T}_2\}$.*

(1) Consider the following two fragments $\mathcal{T}_1$ and $\mathcal{T}_2$ of ontologies that define Cystic_fibrosis_screening. $\mathcal{T}_1$ consists of the definition:

$$\begin{aligned}\text{Cystic\_fibrosis\_screening} \equiv\ & \text{Screening} \sqcap \\ & \exists\text{has\_Focus.Cystic\_fibrosis} \sqcap \\ & \exists\text{has\_Intent.Screening\_procedure\_intent}\end{aligned}$$

and $\mathcal{T}_2$ consists of the inclusions:

$$\begin{aligned}\text{Cystic\_fibrosis\_screening} &\sqsubseteq \text{Genetic\_testing} \\ \text{Genetic\_testing} &\sqsubseteq \text{Molecular\_analysis} \sqcap \text{Screening}.\end{aligned}$$

– Check if $\mathcal{T}_1 \equiv \mathcal{T}_2$?
Answer: No

– For $\Sigma = \{\textsf{Cystic\_fibrosis\_screening}, \textsf{Screening}\}$, check if $\mathcal{T}_1 \equiv_\Sigma \mathcal{T}_2$?
Answer: Yes
Proof: $\forall \mathcal{I}_1|_\Sigma \in \{\mathcal{I}|_\Sigma | \mathcal{I} \models \mathcal{T}_1\}$, we can assume
$Cystic\_fibrosis\_screening^{\mathcal{I}_1} = \{a_1, ..., a_n\}, Screen^{\mathcal{I}_1} = \{a_1, ...a_m\}(m \geq n)$
now we extand $\mathcal{I}_1|_\Sigma$ by setting
$Genetic\_testing^{\mathcal{I}_1} = \{a_1, ...a_n\}, Molecular\_analysis^{\mathcal{I}_1} = \{a_1, ..., a_m\}$
now we have a model of $\mathcal{T}_2$, so $\forall \mathcal{I}_1|_\Sigma \in \{\mathcal{I}|_\Sigma | \mathcal{I} \models \mathcal{T}_1\}, \mathcal{I}_1|_\Sigma \in \{\mathcal{I}|_\Sigma | \mathcal{I} \models \mathcal{T}_2\}$,which means $\{\mathcal{I}|_\Sigma | \mathcal{I} \models \mathcal{T}_1\} \subseteq \{\mathcal{I}|_\Sigma | \mathcal{I} \models \mathcal{T}_2\}$
proving $\{\mathcal{I}|_\Sigma | \mathcal{I} \models \mathcal{T}_2\} \subseteq \{\mathcal{I}|_\Sigma | \mathcal{I} \models \mathcal{T}_1\}$ is similar.

Let $\mathcal{T}_3$ consist of the inclusion:

$$\textsf{Cystic\_fibrosis\_screening} \sqsubseteq \textsf{Screening}.$$

– check if $\mathcal{T}_1 \equiv_\Sigma \mathcal{T}_3$?
Answer: Yes
proof process is similar to the above

– check if $\mathcal{T}_2 \equiv_\Sigma \mathcal{T}_3$?
Answer: Yes
proof process is similar to the above

(2) Assume that a TBox $\mathcal{T}'$ is a *definitorial extension* of a TBox $\mathcal{T}$, i.e., $\mathcal{T}'$ is obtained from $\mathcal{T}$ by adding new concept definitions $A \equiv C$ such that A does neither occur in $\mathcal{T}$ nor on the right-hand side of any of new definitions. For example, suppose that $\mathcal{T}_1$ from above has been extended with:

$$\begin{aligned}\textsf{Tuberculosis\_screening} \equiv\ & \textsf{Bacterial\_disease\_screening} \sqcap \\ & \exists \textsf{has\_Focus.Tuberculosis} \sqcap \\ & \exists \textsf{has\_Intent.Screening\_procedure\_intent},\end{aligned}$$

where Tuberculosis_screening is a new concept name.

– Show that $\mathcal{T} \equiv_\Sigma \mathcal{T}'$ whenever $\mathcal{T}'$ is a definitorial extension of $\mathcal{T}$, for $\Sigma = \textsf{sig}(\mathcal{T})$.
Answer: we need to prove that $\{\mathcal{I}|_\Sigma \mid \mathcal{I} \models \mathcal{T}\} = \{\mathcal{I}|_\Sigma \mid \mathcal{I} \models \mathcal{T}'\}$
first of all,because $\Sigma = sig(\mathcal{T})$, we have $\forall \mathcal{I} \models \mathcal{T}, \mathcal{I}|_\Sigma = \mathcal{I}$
$\forall \mathcal{I}'|_\Sigma \in \{\mathcal{I}|_\Sigma \mid \mathcal{I} \models \mathcal{T}'\} \to \mathcal{I}'|_\Sigma \models \mathcal{T} \to \mathcal{I}'|_\Sigma \in \{\mathcal{I}|_\Sigma \mid \mathcal{I} \models \mathcal{T}\}$,meaning $\{\mathcal{I}|_\Sigma \mid \mathcal{I} \models \mathcal{T}\} \supseteq \{\mathcal{I}|_\Sigma \mid \mathcal{I} \models \mathcal{T}'\}$
now we prove the opposite side
$\{\mathcal{I}|_\Sigma | \mathcal{I} \models \mathcal{T}\}$ consists of all the models of $\mathcal{T}$,so we just extend an

arbitrary model $\mathcal{I}$ of $\mathcal{T}$, let $A^{\mathcal{I}}$(*new concept name*) $= C^{\mathcal{I}}$,then we get a model $\mathcal{I}'$ of $\mathcal{T}'$,and $\mathcal{I}'|_{\Sigma} = \mathcal{I} \rightarrow \mathcal{I} \in \{\mathcal{I}|_{\Sigma}|\mathcal{I} \models \mathcal{T}'\} \rightarrow \{\mathcal{I}|_{\Sigma} \mid \mathcal{I} \models \mathcal{T}\} \subseteq \{\mathcal{I}|_{\Sigma} \mid \mathcal{I} \models \mathcal{T}'\}$
now we have $\{\mathcal{I}|_{\Sigma} \mid \mathcal{I} \models \mathcal{T}\} = \{\mathcal{I}|_{\Sigma} \mid \mathcal{I} \models \mathcal{T}'\}, i.e.\ \mathcal{T} \equiv_{\Sigma} \mathcal{T}'$

- Show that $\Sigma \subseteq \Sigma'$ implies $\equiv_{\Sigma'} \subseteq \equiv_{\Sigma}$, for Σ a signature and Σ' its superset.
  Answer: we need to prove: given $\Sigma \subseteq \Sigma'$,then $\mathcal{T}_1 \equiv_{\Sigma'} \mathcal{T}_2 \rightarrow \mathcal{T}_1 \equiv_{\Sigma} \mathcal{T}_2$
  $\mathcal{T}_1 \equiv_{\Sigma'} \mathcal{T}_2 \rightarrow \{\mathcal{I}|_{\Sigma'}|\mathcal{I} \models \mathcal{T}_1\} = \{\mathcal{I}|_{\Sigma'}|\mathcal{I} \models \mathcal{T}_2\}$
  $\forall \mathcal{I}|_{\Sigma'}$, we can extend it by adding $X^{\mathcal{I}}(X \in \Sigma' \setminus \Sigma)$, then it becomes $\mathcal{I}|_{\Sigma}$, so we have $\{\mathcal{I}|_{\Sigma}|\mathcal{I} \models \mathcal{T}_1\} = \{\mathcal{I}|_{\Sigma}|\mathcal{I} \models \mathcal{T}_2\}$,Q.E.D