

Assignment#4

Yizheng Zhao

May 13, 2020

Note: Assignment#4 contributes to 10% of the total mark of the course.

Deadline: June 2, 2020 before 14:00.

Submission: Submit your work as an A#4-XXX.zip file, for XXX your student ID, via PedagogySquare platform. The zip file should contain a single .pdf file.

Q1. Recall the syntax of the Description Logics \mathcal{EL} , DL-Lite and \mathcal{ALC} . Suppose **Teacher** and **Course** are concept names and **teaches** is a role name.

Let \mathcal{E} be any of the following expressions:

- (a) $\top \sqsubseteq \perp$
- (b) $\exists \text{teaches.Course} \sqsubseteq \text{Teacher}$
- (c) $\forall \text{teaches.Course}$
- (d) $\exists \text{Course.teaches}$
- (e) $\exists \text{teaches}^-. \top \sqsubseteq \text{Teacher} \sqcup \text{School}$
- (f) $\text{Teacher} \sqsubseteq \exists \text{teaches.} \top$
- (g) $\text{Teacher} \sqsubseteq \exists \text{teaches.} \perp$
- (h) $\geq 3 \text{ teaches.} \top \sqsubseteq \text{Teacher}$
- (i) $\geq 4 \text{ teaches.Course} \sqsubseteq \text{Teacher}$
- (j) $\forall \text{teaches.} \top \sqsubseteq \exists \text{teaches.Course}$
- (k) $\exists \text{teaches.} \top \sqsubseteq \geq 2 \text{ teaches.} \top$
- (l) $\geq 2 \text{ teaches.} \top \sqsubseteq \exists \text{teaches.} \top$
- Translate \mathcal{E} into natural language (1 mark per each);
 - (a) Thing belongs to nothing.
 - (b) Object that teaches a course belongs to teacher.
 - (c) Everything that teaches course.
 - (d) This is not a valid concept.

- (e) Object that being taught is either a teacher or a school
 - (f) Teacher is kind of object that teaches something.
 - (g) Teacher is kind of object that teaches nothing.
 - (h) All objects teaches at least 3 things is kind of Teacher.
 - (i) All objects teaches at least 4 courses is kind of Teacher.
 - (j) Everything that teaches something belongs to the object that teaches course.
 - (k) Object that teaches something belongs to the object teaches at least 2 things.
 - (l) Object teaches at least 2 things belongs to the object that teaches things.
- State whether it is (1 mark per each);
 - an \mathcal{EL} concept;
 - an \mathcal{EL} concept inclusion;
 - a DL-Lite concept;
 - a DL-Lite concept inclusion;
 - an \mathcal{ALC} concept;
 - an \mathcal{ALC} concept inclusion;
 - none of the above.
 - (a) an \mathcal{ALC} concept inclusion and a DL-Lite concept inclusion;
 - (b) an \mathcal{EL} concept inclusion and an \mathcal{ALC} concept inclusion;
 - (c) an \mathcal{ALC} concept;
 - (d) none of the above;
 - (e) none of the above;
 - (f) an \mathcal{EL} concept inclusion and an \mathcal{ALC} concept inclusion and a DL-Lite concept inclusion;
 - (g) an \mathcal{ALC} concept inclusion;
 - (h) a DL-Lite concept inclusion;
 - (i) none of the above;
 - (j) an \mathcal{ALC} concept inclusion;
 - (k) a DL-Lite concept inclusion;
 - (l) a DL-Lite concept inclusion;
- If \mathcal{E} is a concept inclusion, check whether \mathcal{E} follows from the empty TBox (i.e., $\emptyset \models \mathcal{E}$). If this is not the case, define an interpretation \mathcal{I} such that $\mathcal{I} \not\models \mathcal{E}$ (1 mark per each)

- (a) \mathcal{E} not follows from the empty TBox. We can make $\top^{\mathcal{I}} = \{a\}, \perp^{\mathcal{I}} = \emptyset$.
 - (b) \mathcal{E} not follows from the empty TBox. We can make $\Delta^{\mathcal{I}} = \{x, y, z\}, Teacher^{\mathcal{I}} = \{x\}, Course^{\mathcal{I}} = \{y, z\}, teaches^{\mathcal{I}} = \{(z, y)\}$.
 - (c) \mathcal{E} not follows from the empty TBox. We can make $\Delta^{\mathcal{I}} = \{x, y, z\}, Teacher^{\mathcal{I}} = \{z\}, Course^{\mathcal{I}} = \{y\}, School^{\mathcal{I}} = \{y\}, teaches^{\mathcal{I}} = \{(z, y)\}$.
 - (d) \mathcal{E} not follows from the empty TBox. We can make $\Delta^{\mathcal{I}} = \{x, y, z\}, Teacher^{\mathcal{I}} = \{x\}, Course^{\mathcal{I}} = \{y, z\}, teaches^{\mathcal{I}} = \{(z, y)\}$.
 - (e) \mathcal{E} not follows from the empty TBox. We can make $\Delta^{\mathcal{I}} = \{x, y\}, Teacher^{\mathcal{I}} = \{x\}, teaches^{\mathcal{I}} = \{(y, \emptyset)\}$.
 - (f) \mathcal{E} not follows from the empty TBox. We can make $\Delta^{\mathcal{I}} = \{x, y, z, i, j\}, Teacher^{\mathcal{I}} = \{x\}, \neg Teacher^{\mathcal{I}} = \{y, z, i, j\}, teaches^{\mathcal{I}} = \{(y, z), (y, i), (y, j)\}$.
 - (g) \mathcal{E} not follows from the empty TBox. We can make $\Delta^{\mathcal{I}} = \{x, y, z, i, j, k\}, Teacher^{\mathcal{I}} = \{x\}, \neg Teacher^{\mathcal{I}} = \{y, z, i, j, k\}, Course^{\mathcal{I}} = \{z, i, j, k\}, teaches^{\mathcal{I}} = \{(y, z), (y, i), (y, j), (y, k)\}$.
 - (h) \mathcal{E} not follows from the empty TBox. We can make $\Delta^{\mathcal{I}} = \{x, y, z, k\}, Course^{\mathcal{I}} = \{z\}, teaches^{\mathcal{I}} = \{(x, z), (k, y)\}$.
 - (i) \mathcal{E} not follows from the empty TBox. We can make $\Delta^{\mathcal{I}} = \{x, y, z, k\}, teaches^{\mathcal{I}} = \{(x, z), (x, y), (k, z)\}$.
 - (j) \mathcal{E} follows from the empty TBox.
- If \mathcal{E} is a concept, check whether \mathcal{E} is satisfiable. If this is the case, define an interpretation \mathcal{I} such that $\mathcal{E}^{\mathcal{I}} \neq \emptyset$ (1 mark per each).
 - (c) \mathcal{E} is satisfiable. We can make $\Delta^{\mathcal{I}} = \{x, y, z\}, Course^{\mathcal{I}} = \{y, z\}, teaches^{\mathcal{I}} = \{(x, z), (x, y)\}$.
 - (d) This is not a valid concept.

Q2. Show that every \mathcal{EL} -TBox is satisfiable (consistent). That is, show that for every \mathcal{EL} -TBox \mathcal{T} there exists an interpretation \mathcal{I} such that $\mathcal{I} \models \mathcal{T}$ (10 marks).

Assume that the \mathcal{EL} -TBox is in normal form, for every concept inclusions $A \sqsubseteq B$ in \mathcal{EL} -TBox, we can constructively make an interpretation \mathcal{I} such that $A^{\mathcal{I}} \sqsubseteq B^{\mathcal{I}}$. Therefore, $\mathcal{I} \models \mathcal{T}$, which means every \mathcal{EL} -TBox is satisfiable.

Q3. Consider the database instance $\mathcal{D}_{\text{PGBBW}}$ (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, slowly)  hasFriend(tibbie, jonie)
hasEnemy(paddi, wolffy)  hasWife(wolffy, wolnie)  hasSon(wolnie, wilie)

```

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

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

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

- (a) $\text{Sheep}(\text{fitty})$
- (b) $\text{Sheep}(\text{wolnie})$
- (c) $\text{Sheep}(\text{paddi})$
- (d) $\neg\text{Sheep}(\text{paddi})$
- (e) $(\exists\text{hasFriend}.\top)(\text{weslie})$
- (f) $(\exists\text{hasFriend}.\text{Sheep})(\text{weslie})$
- (g) $(\exists\text{hasFriend}.\text{LazySheep})(\text{weslie})$
- (h) $(\text{BrownSheep} \sqcap \neg\text{LazySheep})(\text{fitty})$
- (i) $(\text{BrownSheep} \sqcap \neg\text{Sheep})(\text{fitty})$
- (j) $\text{Sheep}(\text{wilie})$
- (k) $(\exists\text{hasSon}.\neg\text{Sheep})(\text{wolnie})$
- (l) $(\exists\text{hasEnemy}.\exists\text{hasWife}.\text{Wolf})(\text{paddi})$

- 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) (1 mark per each).

- (a) $\text{Sheep}(\text{fitty})$

- (b) `Sheep(wolnie)`
 - (c) `Sheep(paddi)`
 - (d) `¬Sheep(paddi)`
 - (e) `∃x.hasFriend(weslie,x)`
 - (f) `∃x.(hasFriend(weslie,x) ∧ Sheep(x))`
 - (g) `∃x.(hasFriend(weslie,x) ∧ LazySheep(x))`
 - (h) `(BrownSheep ∧ ¬LazySheep)(fitty)`
 - (i) `(BrownSheep ∧ ¬Sheep)(fitty)`
 - (j) `Sheep(wilie)`
 - (k) `∃x.(hasSon(wolnie,x) ∧ ¬Sheep(x))`
 - (l) `∃x.(hasEnemy(paddi,x) ∧ ∃y.(hasWife(x,y) ∧ Wolf(y)))`
- Query answering under closed world assumption: check for each Boolean F whether the answer to the query F given by $\mathcal{D}_{\text{PGBBW}}$ is “Yes” or “No”. In other words, check whether $\mathcal{I} \models F$ or $\mathcal{I} \models \neg F$ (1 mark per each).
 - (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 $\mathcal{D}_{\text{PGBBW}}$ 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 (1 mark per each).
 - (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) $F_1(x) = \text{Wolf}(x)$
- (b) $F_2(x) = \neg \text{Sheep}(x)$
- (c) $F_3(x, y) = \text{hasFriend}(x, y)$
- (d) $F_4(x) = \text{Sheep}(x) \wedge \neg \text{hasFriend}(x, \text{jonie})$

For each query F_i , give

- for closed world assumption: $\text{answer}(F_i, \mathcal{D}_{\text{PGBBW}})$ (1 mark per each);
 - (a) $\text{answer}(F_1, \mathcal{D}_{\text{PGBBW}}) = \{\text{wolffy, wolnie, wilie}\}$
 - (b) $\text{answer}(F_2, \mathcal{D}_{\text{PGBBW}}) = \{\text{paddi, fitty, wolffy, wolnie, wilie}\}$
 - (c) $\text{answer}(F_3, \mathcal{D}_{\text{PGBBW}}) = \{(\text{weslie, slowy}), (\text{tibbie, jonie})\}$
 - (d) $\text{answer}(F_4, \mathcal{D}_{\text{PGBBW}}) = \{\text{weslie, slowy, jonie}\}$
- for open world assumption: $\text{certanswer}(F_i, \mathcal{D}_{\text{PGBBW}})$ (1 mark per each).
 - (a) $\text{certanswer}(F_1, \mathcal{D}_{\text{PGBBW}}) = \{\text{wolffy, wolnie, wilie}\}$
 - (b) $\text{certanswer}(F_2, \mathcal{D}_{\text{PGBBW}}) = \emptyset$
 - (c) $\text{certanswer}(F_3, \mathcal{D}_{\text{PGBBW}}) = \{(\text{weslie, slowy}), (\text{tibbie, jonie})\}$
 - (d) $\text{certanswer}(F_4, \mathcal{D}_{\text{PGBBW}}) = \emptyset$

Q4. Clarify, in your own words, the distinctions between closed world assumption (CWA) and open world assumption (OWA). When do CWA and OWA apply? Illustrate with an example how CWA and OWA lead to a distinction when applying them to a data/knowledge base (approx. 500 words, 10 marks).

The most important difference between CWA and OWA is that CWA thinks that things that are not stated must be false and OWA assumes that in this case the answer is "don't know". This means, in CWA, if a thing is not known to us, then it is false; in OWA, it may be true or false which means we don't know.

When we are building a database or knowledge base, if we have detailed information about the system, then CWA should be applied. For example, there is a student score management program with a student A in the database but A do not have ICS (a course in NJU) score. If we query "Student A's ICS score is 99", under CWA, it will return false, because student A's ICS score is not stated so it is false, but under OWA, it will return "don't know". According to our needs, obviously when a student does not have a grade in a certain course, it means that he or she does not take such a course, then query his grade in this course is naturally false. From the perspective of school, the school has complete information about the students, so this database should use CWA.

On the other hand, when we don't have complete information about the system, OWA should be applied. For example, we have a knowledge base about physics, we have a query "Quantum mechanics and general relativity can be unified". Under CWA, it will return false because at present we do not know whether quantum mechanics and general relativity can be unified. Under OWA, it will return "don't know". As we know, knowledge is not immutable, but will change or be corrected. Right now we are not sure about the relation between quantum mechanics and general relativity, so we should output "don't know". So in this case, knowledge base should apply OWA.

Let's analyze the specific differences between OWA and CWA further by introducing another example. When making a sushi ontology, it is assumed that tomato can only be one type of thing. Now we have tomato is subsumed by vegetables and tomato is subsumed by fruit. Under CWA, an error will be returned because it will assume that vegetable and fruit are different things since they are listed. Under OWA, no error will occur, because it will assume vegetable is equivalent to fruit since tomato can only be one type of thing.