KRP	**Spring 2023**

Homework 1

Instructor: YiZheng Zhao *Name:* , *StudentId:* 211300063

Question 1. Basic Understanding of KR and Ontologies

A Logic-based ontology is a formal, explicit specification of a shared conceptualization and has decidability and rationality. For areas of interest based on a fixed vocabulary, knowledge can be expressed in terms of finite words and then determined and reasoned by the formal logic syntax.For example, we can use syllogism to reason more knowledge.

If we interpret domains and concepts as sets and elements of sets,the KR model can be calculated semantically using set theory. We can also interpret some logical words as sets operations, such as unions, intersections, and subsets, are also computational.

Question 2. Expressivity & Computability

Example: Has Mary eaten yet?

My opinion: Expressiveness and computability are trade-offs.The more expressive a logic-based KR language is, the weaker it may be in some ways, such as completeness, soundness and decidability. For example, FOL is more expressive than ALC, but FOL is undecided and decidabile while ALC is decidabile and computabile.More than that, Natural language is as expressive as possible in any circumstances, but it is not a computational model. Therefore, a logic-based KR language is not more expressive, more suitable for knowledge representation, but also to examine its computability.

Question 3. Manchester Syntax

(1) Timo is a cow.

(2) No.

(3) Not enough.

(4) zero.

Question 4. ALC Extensions & FOL

(1)(1.1) Sentence: Every Chinese couple has at most 3 children.
Concept names: ChineseCouple
Role names: hasChildren
Inclusion: ChineseCouple $\sqsubseteq$ ($\leq$ 3 hasChildren.$\top$)

(1.2) Sentence: ML is a course taught by SFM who is a professor working at NJU.
Concept names: Course, Professor
Role names: taughtBy, workingAt
Nominals: ML, SFM, NJU
Inclusion: {ML} $\sqsubseteq$ Course $\sqcap$ ($\exists$taughtBy.{SFM})
{SFM} $\sqsubseteq$ Professor $\sqcap$ ($\exists$ workingAt.{NJU})

(1.3) Sentence: NJU is a university whose members are a school or a department.
Concept names: University, School, Department
Role names: hasMember
Nominals: NJU
Inclusion: {NJU} $\sqsubseteq$ University $\sqcap$($\exists$ hasMember.$\top$)$\sqcap$($\forall$ hasMember.(School$\sqcup$ Department))

(1.4) Sentence: NJU has at least 30,000 students.
Concept names: Students
Role names: has
Nominals: NJU
Inclusion: {NJU} $\sqsubseteq$ ($\geq 30,000$ has.Student)

(1.5) Sentence: All members of AI School are undergraduates, graduates, or teachers.
Concept names: Undergraduates, Graduates, Teacher
Role names: memberOf
Nominals: AI school
Inclusion: $\forall$ memberOf.{AI school} $\sqsubseteq$ Undergraduate$\sqcup$Graduate$\sqcup$Teacher

(1.6) Sentence: The domain of the relation "citizenOf" consists of countries.
Concept names: Country
Role names: citizenOf
Inclusion: $\exists$ citizenOf$^{-}.\top \sqsubseteq$ Country

(2)(2.1) $\forall x(\text{memberOf}(x, \text{AI School}) \rightarrow \text{Undergraduate}(x) \vee \text{Graduate}(x) \vee \text{Teacher}(x))$

(2.2) $\forall x(\exists y(\text{citizenOf}(y, x)) \rightarrow \text{Country}(x))$

Question 5. DL Semantics

(1) True. We only need one example to prove it.
Ontology: $\top \sqsubseteq \bot$, with no empty domain $\Delta^{\mathcal{I}}$.

(2) False.
If there is an ontology has finite models, we can always replace an element in the domain with a new element (which is not in the original domain), so we get a new model. Because the new elements are infinite, we can construct an infinite number of models.

(3) True.
Based on the results of the previous two questions, obviously every ontology has either no model or infinite many models.

(4) True.
According to the def of satisfiability, a class C is satisfiable w.r.t $\mathcal{T}$ iff $C^I \neq \emptyset$ for some model $\mathcal{I}$ of $\mathcal{T}$.
$\because C^I \neq \emptyset \therefore \mathcal{I}$ is a non-empty interpretation of C.

(5) False.
If there is an unsatisfied class in a model with a non-empty interpretation,

then it also satisfies the definition of satisfiable class that conflicts with the original statement.

(6) True.
According to the result of (5), we know that an unsatisfiable class is interpreted as an empty set($C^I = \emptyset$), so it will be a subclass of any other class.

Question 6. Interpretation as Graph

(1) $(\neg A)^{\mathcal{I}} = \{f, h, i\}$

(2) $(\exists r.(A \sqcup B))^{\mathcal{I}} = \{d, f\}$

(3) $(\exists s.\exists s.\neg A)^{\mathcal{I}} = \{d, e\}$

(4) $(\neg A \sqcap \neg B)^{\mathcal{I}} = \{f, h, i\}$

(5) $(\forall r.(A \sqcup B))^{\mathcal{I}} = \{d, f, g, h, i\}$

(6) $(\leq 1s.\top)^{\mathcal{I}} = \{e, f, g, h, i\}$

Question 7. DL Semantics

(1) $(Q \sqcap \geq 2r.P)^{\mathcal{I}} = \emptyset$
$(\forall r.Q)^{\mathcal{I}} = \{b, c, d, e\}$
$(\neg \exists r.Q)^{\mathcal{I}} = \{b, c, e\}$
$(\forall r.\top \sqcap \exists r^{-}.P)^{\mathcal{I}} = \{b, d, e\}$
$(\exists r^{-}.\bot)^{\mathcal{I}} = \emptyset$

(2)(2.1) $(A \sqcap B)^{\mathcal{I}} = \emptyset$
$(\exists r.B)^{\mathcal{I}} = \{1, 2\}$
$(\exists r.(A \sqcap B))^{\mathcal{I}} = \emptyset$
$\top^{\mathcal{I}} = \{1, 2, 3, 4, 5, 6\}$
$(A \sqcap \exists r.B)^{\mathcal{I}} = \{1, 2\}$

(2.2) $\mathcal{I} \models A \equiv \exists r.B$
$\mathcal{I} \models A \cap B \sqsubseteq \top$
$\mathcal{I} \models \exists r.A \sqsubseteq A \sqcap B$

Question 8. DL Semantics

(1) True.
$\because C \sqsubseteq D \therefore C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$for all models of $C \sqsubseteq D$
which means: $\forall x \in \Delta^{\mathcal{I}}$, if x $\in C^{\mathcal{I}}$, then x $\in D^{\mathcal{I}}$

$$\begin{aligned}(\exists r.C)^{\mathcal{I}} &= \{x \in \Delta^{\mathcal{I}} | \exists y \in \Delta^{\mathcal{I}} \text{s.t.} (x,y) \in r^{\mathcal{I}} \text{ and } y \in C^{\mathcal{I}}\} \\ &\subseteq \{x \in \Delta^{\mathcal{I}} | \exists y \in \Delta^{\mathcal{I}} \text{s.t.} (x,y) \in r^{\mathcal{I}} \text{ and } y \in D^{\mathcal{I}}\} \\ &= (\exists r.D)^{\mathcal{I}}\end{aligned}$$

$\therefore \exists r.C \sqsubseteq \exists r.D$ holds.(completeness of DL)

(2) False.
A counterexample:
$\Delta^{\mathcal{I}} = \{a, b, c\}$
$C^{\mathcal{I}} = \{c\}$
$r^{\mathcal{I}} = \{(a,a), (b,b)\}$
$\therefore (\exists r.C)^{\mathcal{I}} = \emptyset, \leq 1r.\top = \{a, b\}$
$(\exists r.C)^{\mathcal{I}} \neq (\leq 1r.\top)^{\mathcal{I}}$
$\therefore \exists r.C$ is not equivalent to $\leq 1r.\top$(completeness of DL)

(3) True.
For any models $\mathcal{I}$,

$$\begin{aligned}(\leq 0r.\top)^{\mathcal{I}} &= \{x \in \Delta^{\mathcal{I}} | \text{card}\left(\{y \in \Delta^{\mathcal{I}} | (x,y) \in r^{\mathcal{I}} \text{ and } y \in \top^{\mathcal{I}}\}\right) \leq 0\} \\ &= \{x \in \Delta^{\mathcal{I}} | \{y \in \Delta^{\mathcal{I}} | (x,y) \in r^{\mathcal{I}} \text{ and } y \in \Delta^{\mathcal{I}}\} = \emptyset\} \qquad (1) \\ &= \{x \in \Delta^{\mathcal{I}} | \{y \in \Delta^{\mathcal{I}} | (x,y) \in r^{\mathcal{I}}\} = \emptyset\}\end{aligned}$$

$$\begin{aligned}(\forall r.\bot)^{\mathcal{I}} &= \{x \in \Delta^{\mathcal{I}} | \forall y \in \Delta^{\mathcal{I}} \text{ if } (x,y) \in r^{\mathcal{I}} \text{ then } y \in \bot^{\mathcal{I}}\} \\ &= \{x \in \Delta^{\mathcal{I}} | \forall y \in \Delta^{\mathcal{I}} \text{ if } (x,y) \in r^{\mathcal{I}} \text{ then } y \in \emptyset\} \qquad (2) \\ &= \{x \in \Delta^{\mathcal{I}} | \{y \in \Delta^{\mathcal{I}} | (x,y) \in r^{\mathcal{I}}\} = \emptyset\}\end{aligned}$$

$\therefore (\leq 0r.\top)^{\mathcal{I}} = (\forall r.\bot)^{\mathcal{I}}$
Because of completeness of DL, we can know that$\leq 0r.\top$ is equivalent to $\forall r.\bot$

(4) False.
A counterexample:
$\Delta^{\mathcal{I}} = \{a, b, c\}$

$A^{\mathcal{I}} = \{a, c\}$
$B^{\mathcal{I}} = \{b\}$
$r^{\mathcal{I}} = \{(a, b), (a, c)\}$
$\therefore (\forall r.(A \sqcup B))^{\mathcal{I}} = \{a, b, c\}, ((\forall r.A) \sqcup (\forall r.B))^{\mathcal{I}} = \{b, c\}$
$(\forall r.(A \sqcup B))^{\mathcal{I}} \neq ((\forall r.A) \sqcup (\forall r.B))^{\mathcal{I}}$
$\therefore \forall r.(A \sqcup B)$ is not equivalent to $(\forall r.A) \sqcup (\forall r.B)$.(completeness of DL)

(5) True.
"$\Longrightarrow$":
$\forall a \in (\exists r.(A \sqcup B))^{\mathcal{I}}, \exists (a, b) \in r^{\mathcal{I}}$ and $(b \in A^{\mathcal{I}}$ or $b \in B^{\mathcal{I}})$
$\therefore a \in ((\exists r.A) \sqcup (\exists r.B))^{\mathcal{I}}$
"$\Longleftarrow$":
$\forall c \in ((\exists r.A) \sqcup (\exists r.B))^{\mathcal{I}}, \exists d \in A^{\mathcal{I}}$ or $B^{\mathcal{I}}$ s.t. $(c, d) \in r^{\mathcal{I}}$
$\therefore c \in (\exists r.(A \sqcup B))^{\mathcal{I}}$
$\therefore (\exists r.(A \sqcup B))^{\mathcal{I}} = ((\exists r.A) \sqcup (\exists r.B))^{\mathcal{I}}$
$\therefore \exists r.(A \sqcup B)$is equivalent to $(\exists r.A) \sqcup (\exists r.B)$.(completeness of DL)

Question 9. DL Semantics

$\Delta^{\mathcal{I}} = \{a, b, c\}$
$Person^{\mathcal{I}} = \{a, b, c\}$
$Parent^{\mathcal{I}} = \{a, b\}$
$Mother^{\mathcal{I}} = \{a\}$
$hasChildren^{\mathcal{I}} = \{(a, c), (b, c)\}$
$\therefore \mathcal{I} \models \mathcal{T}, \mathcal{I} \not\models$ Parent $\sqsubseteq$ Mother.

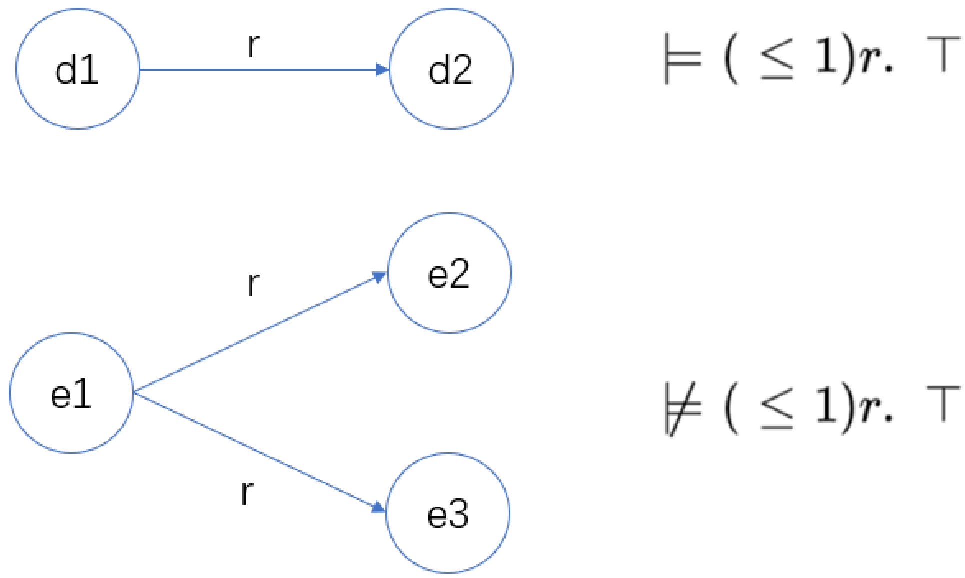
Question 10. DL Semantics

(1) $\Longrightarrow$:
According to the def of subsumption, $X \sqsubseteq_{\mathcal{T}} Y$ iff $X^{\mathcal{I}} \subseteq Y^{\mathcal{I}}$ for all models $\mathcal{I}$ of $\mathcal{T}$.
$\therefore X^{\mathcal{I}} \cap \neg Y^{\mathcal{I}} = \emptyset$ for all models $\mathcal{I}$ of $\mathcal{T}$.
$\therefore X \sqcap \neg Y$ is not satisfibale with respect to $\mathcal{T}$.
$\Longleftarrow$:
if $X \sqcap \neg Y$ is not satisfibale with respect to $\mathcal{T}$, then for all models $\mathcal{I}$ of $\mathcal{T}$, we have $(X \sqcap \neg Y)^{\mathcal{I}} = \emptyset$

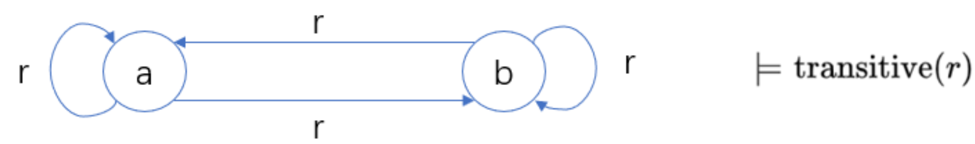
$\therefore X^{\mathcal{I}} \cap \neg Y^{\mathcal{I}} = \emptyset$ for all models $\mathcal{I}$ of $\mathcal{T}$
$X^{\mathcal{I}} \subseteq Y^{\mathcal{I}}$for all models $\mathcal{I}$ of $\mathcal{T}$
According to the def of subsumption, $X \sqsubseteq_{\mathcal{T}} Y$.

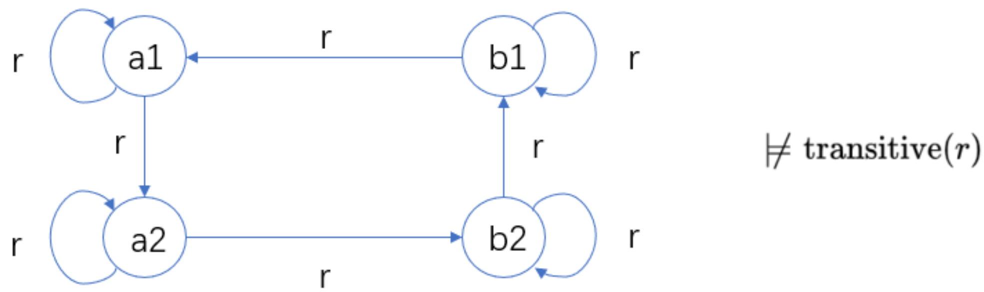
(2) According to the result of (1), let $Y = \bot$, we have $X \sqsubseteq_{\mathcal{T}} \bot$ iff $X \sqcap \neg\bot = X$ is not satisfibale with respect to $\mathcal{T}$.
The inverse negation of the above inference is the conclusion we want to prove: X is satisfiable with respect to $\mathcal{T}$ iff $X \not\sqsubseteq \bot$.

Question 11. Bisimulation invariance



(1) As shown in the picture above, $(\mathcal{I}_1, d_1) \sim (\mathcal{I}_2, e_1)$, so there is not a ALC-Concept could distinguish d_1 and e_1, but ALCQ-Concept $(\leq 1)r.\top$ can distinguish d_1 and e_1.
Therefore $\mathcal{ALCQ}$ is more expressive than $\mathcal{ALC}$.



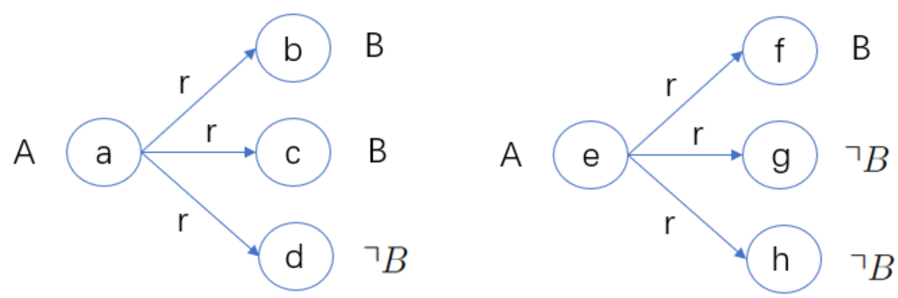


(2) As shown in the picture above, $\mathcal{I}_1 \sim \mathcal{I}_2$, so there is not a $\mathcal{ALC}$-Concept could distinguish the two interpretations, but $\mathcal{S}$-Concept $transitive(r)$ can distinguish $\mathcal{I}_1$ and $\mathcal{I}_2$.
Therefore $\mathcal{S}$ is more expressive than $\mathcal{ALC}$.

Question 12. Bisimulation invariance

(1) Definition of $\mathcal{ALCN}$-bisimulation:
let $\mathcal{I}_1$ and $\mathcal{I}_2$ be Interpretations. The relation $\rho \subset \Delta^{\mathcal{I}_1} \times \Delta^{\mathcal{I}_2}$ is a bisimulation between $\mathcal{I}_1$ and $\mathcal{I}_2$ iff:
a. $d_1 \rho d_2$ implies $d_1 \in A^{\mathcal{I}_1}$ iff $d_2 \in A^{\mathcal{I}_2}$ for all $A \in \mathbf{C}$
b. $d_1 \rho d_2$ and $(d_1, d_1^{'}) \in r^{\mathcal{I}_1}$ implies the existence of $d_2^{'} \in \Delta^{\mathcal{I}_2}$ such that $d_1^{'} \rho d_2^{'}$ and $(d_2, d_2^{'}) \in r^{\mathcal{I}_2}$ for all $r \in \mathbf{R}$
c. $d_1 \rho d_2$ and $(d_2, d_2^{'}) \in r^{\mathcal{I}_2}$ implies the existence of $d_a^{'} \in \Delta^{\mathcal{I}_1}$ such that $d_1^{'} \rho d_2^{'}$ and $(d_1, d_1^{'}) \in r^{\mathcal{I}_1}$ for all $r \in \mathbf{R}$
d. $d_1 \rho d_2$ and card$\left(\{d \in \Delta^{\mathcal{I}_1} | (d_1, d) \in r^{\mathcal{I}_1}\}\right) = n (n \in \mathbf{Z})$ implies card$\left(\{d \in \Delta^{\mathcal{I}_2} | (d_2, d) \in r^{\mathcal{I}_2}\}\right) = n$ for all $r \in \mathbf{R}$
e. $d_1 \rho d_2$ and card$\left(\{d \in \Delta^{\mathcal{I}_2} | (d_2, d) \in r^{\mathcal{I}_2}\}\right) = n (n \in \mathbf{Z})$ implies card$\left(\{d \in \Delta^{\mathcal{I}_1} | (d_1, d) \in r^{\mathcal{I}_1}\}\right) = n$ for all $r \in \mathbf{R}$

(2) Firstly, we need to prove the bisimulation invariance of $\mathcal{ALCQ}$.We just have to prove the case when $C = (\leq n)r.$, other cases are as the same as $\mathcal{ALC}$.
let: $C = (\leq n)r.$, if $d_1 \in C^{\mathcal{I}_1}$, then card$\left(\left\{d \in \Delta^{\mathcal{I}_1} | (d_1, d) \in r^{\mathcal{I}_1}\right\}\right) \leq n$, if $(\mathcal{I}_1, d_1) \sim (\mathcal{I}_2, d_2)$, then card$\left(\left\{d \in \Delta^{\mathcal{I}_1} | (d_2, d) \in r^{\mathcal{I}_1}\right\}\right) \leq n$, so $d_2 \in C^{\mathcal{I}_1}$
Secondly, we show that there exists models which $\mathcal{ALCN}$ could not distinguish but $\mathcal{ALCQ}$ could do that.
According to the def of bisimulation of $\mathcal{ALCN}$, $(\mathcal{I}_1, a) \sim (\mathcal{I}_2, e)$, so there



is not a $\mathcal{ALCN}$-Concept could distinguish a and e, but $\mathcal{ALCN}$-Concept $(\leq 1)r.\neg B$ can distinguish d_1 and e_1.
Therefore $\mathcal{ALCQ}$ is more expressive than $\mathcal{ALCN}$.

Question 13. Make the acquaintance of Prote´ge´

(1) axiom count: 801; logical axiom count: 322
They are different because axiom consists of logical axioms and non-logic axioms. Logic axioms are the axioms that could be used to reasoning, while non-axioms do not have the function, such as declaration axioms and annotation assertions.

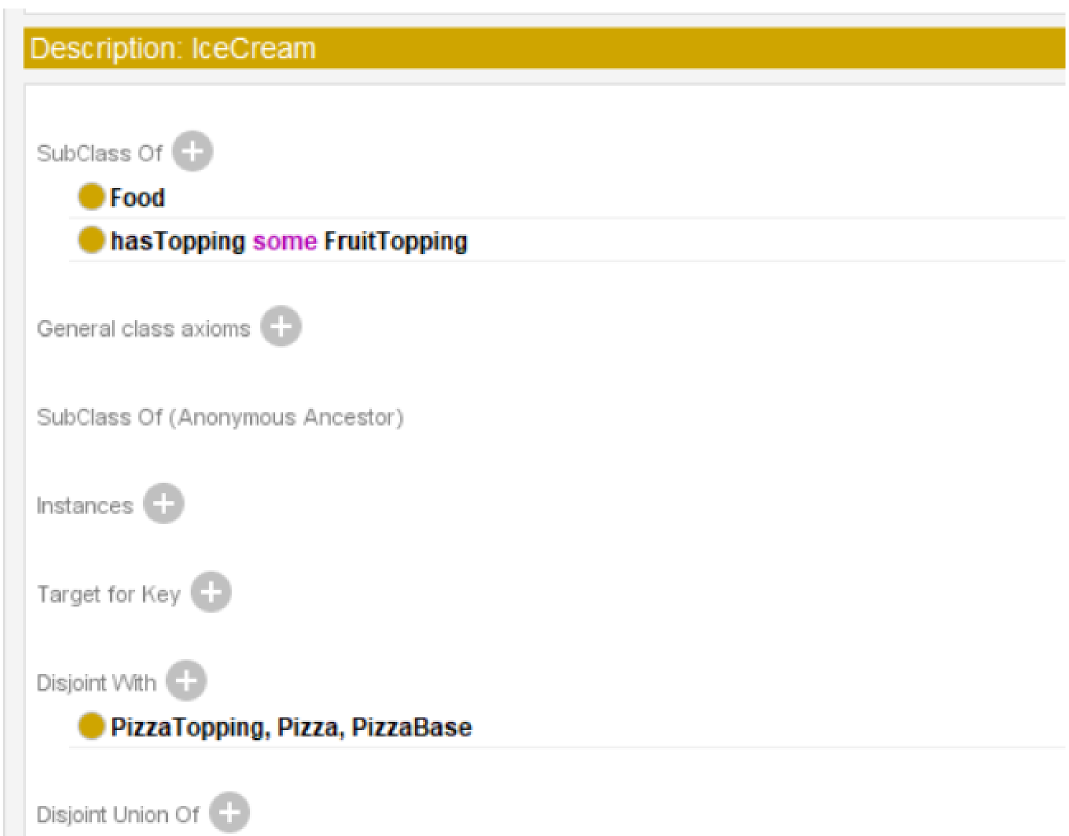
(2)(1.1) use nominals: Country is EquivalentTo DomainThing and (America, England, France, Germany, Italy)

(1.2) use nogations:
VegetarianPizza EquivalentTo Pizza and (not (hasTopping some Seafood-Topping)) and (not (hasTopping some MeatTopping))

NonVegetarianPizza EquivalentTo Pizza and (not VegetarianPizza)

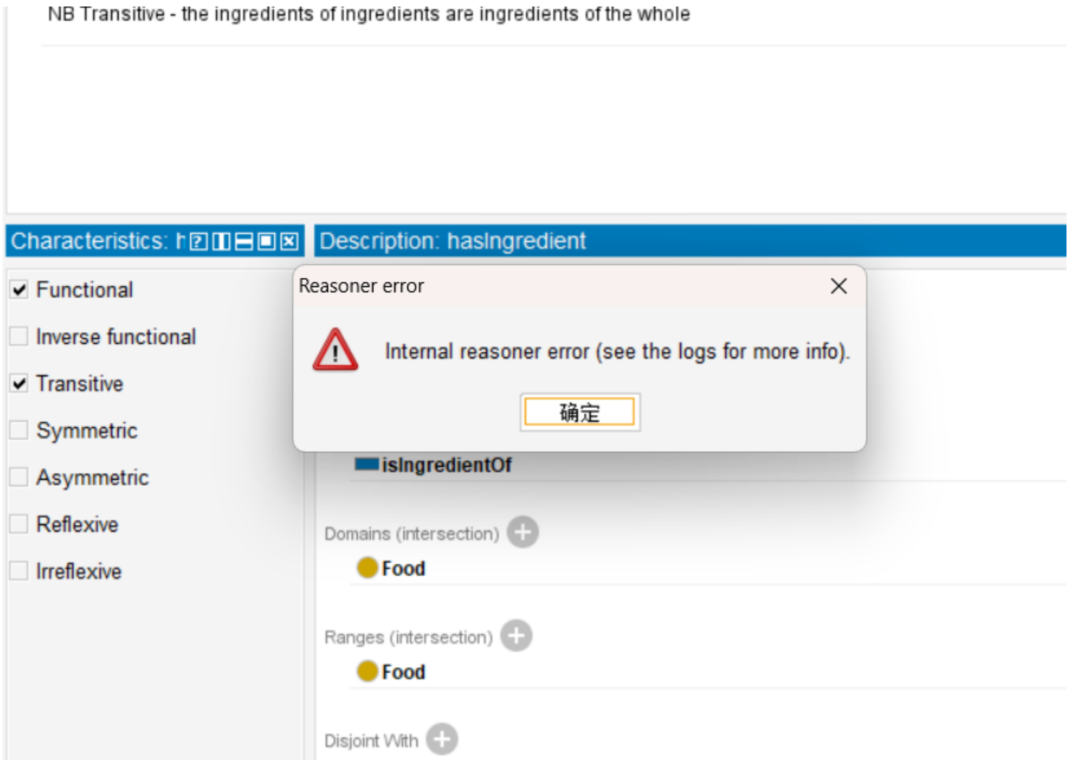
(1.3) declare a sub-property of an object property:
hasTopping SubPropertyOf: hasIngredient
isBaseOf SubPropertyOf: isIngredientOf
isToppingOf SubPropertyOf: isIngredientOf
hasBase SubPropertyOf: hasIngredient

(1.4) declare an inverse property:
hasBase InverseOf isBaseOf
hasTopping InverseOf isToppingOf
hasIngredient InverseOf isIngredientOf



(3) Because IceCream is equivalent to Nothing. IceCream has a restriction "SubClassOf hasTopping some FruitTopping", but hasTopping has a domain of Pizza, which is disjoint with IceCream(we can see this in the picture), So there's a contradiction.

(4) class ::: superclass
CajunSpiceTopping: SpicyTopping.
SloppyGiuseppe: CheesyPizza, InterestingPizza, MeatyPizza, SpicyPizza, SpicyPizzaEquivalent.



(5) It will throw a warning(as the picture). Because there are some food have various ingredient, so the object property "hasIngredient" is not a function.

Question 14. Develop your first ontology with Prote´ge´

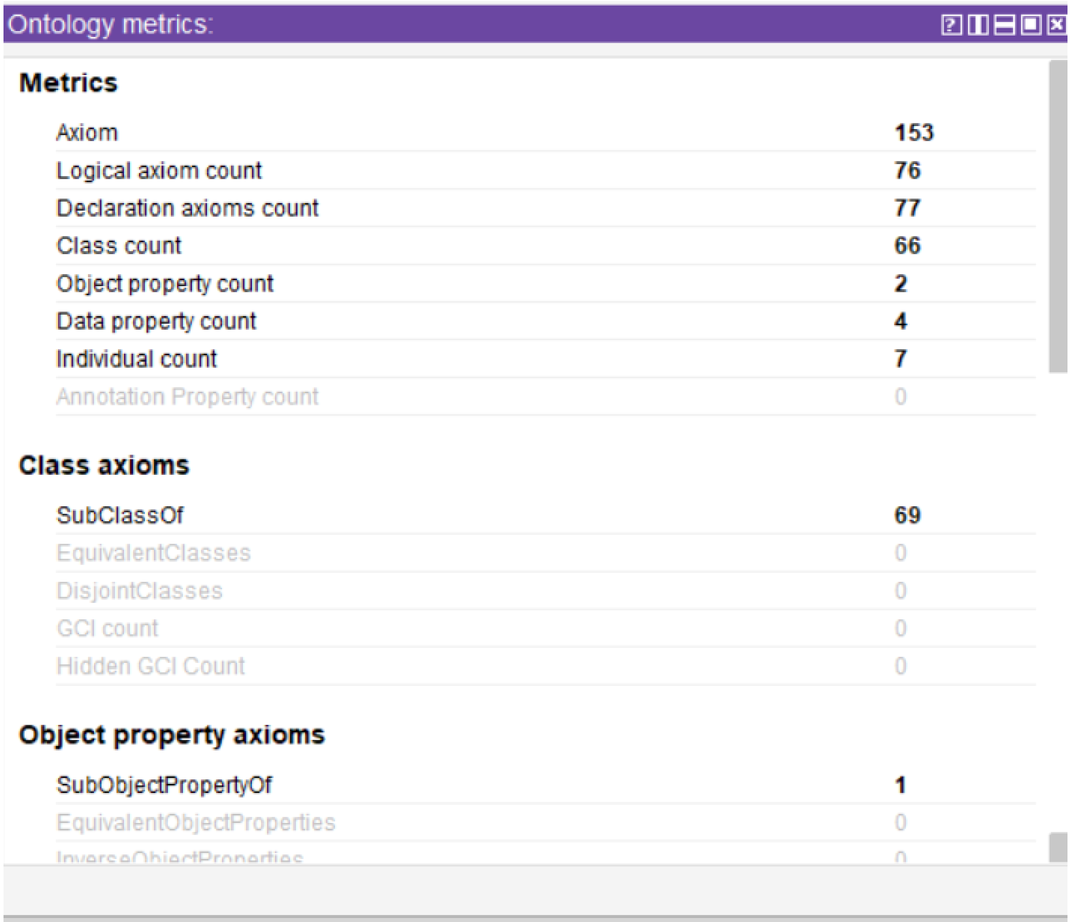


Figure 1: Ontology metrics

As the picture show, I built a hierarchy of sports, activity and place(Figure 2) I built a object properties to represent the stadium of a sport(Figure 3), I built two data properties to represent the begin time and end time of each sport(Figure 4), I built a data property to represent whether a game produces a gold medal(Figure 5).

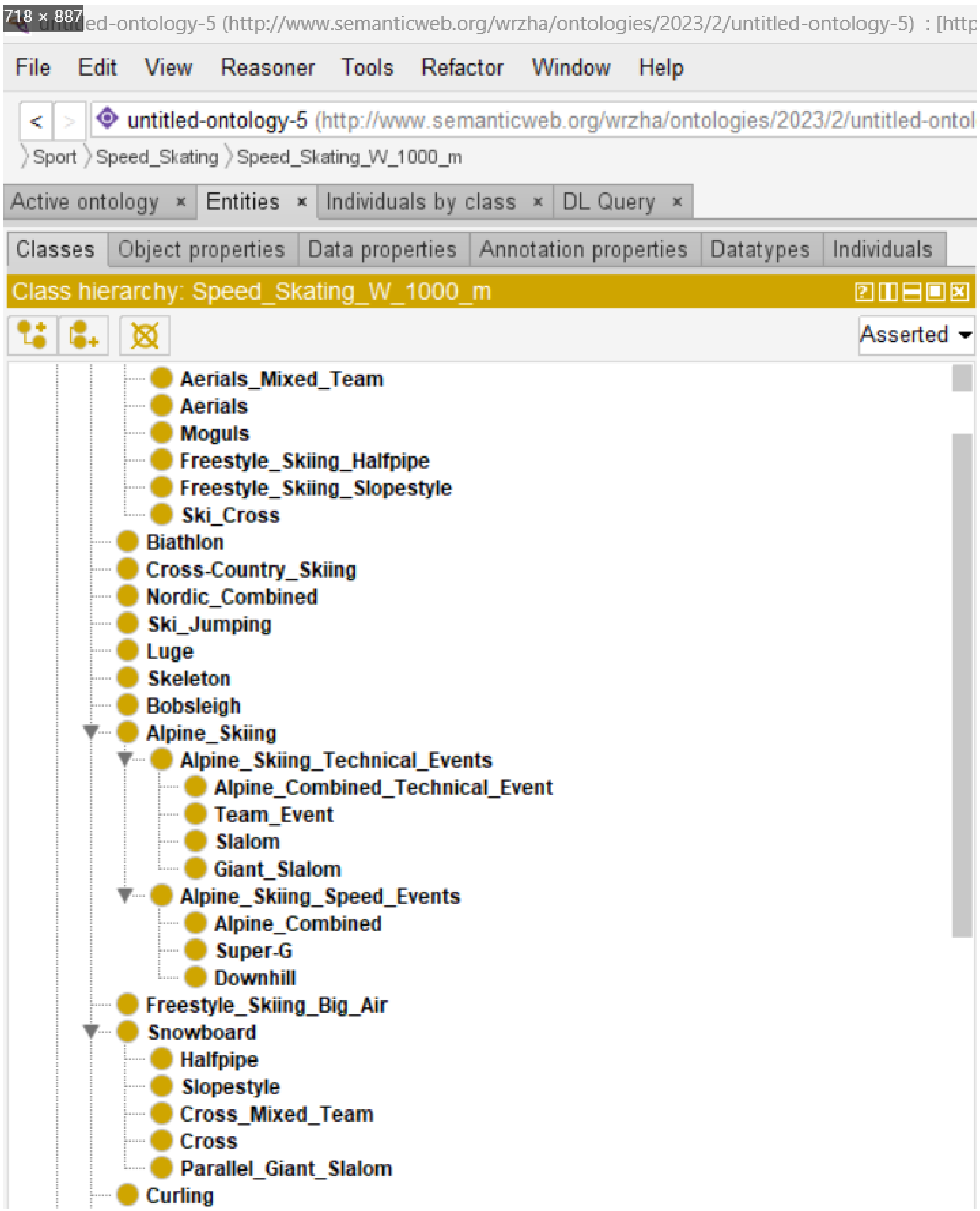


Figure 2: Hierarchy of sports items

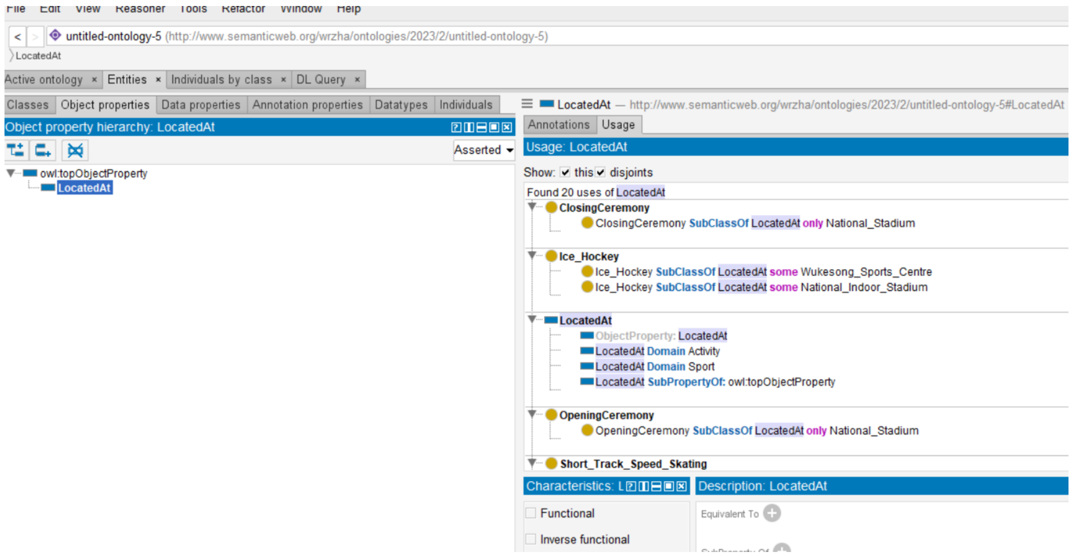


Figure 3: Object properties: LocatedAt

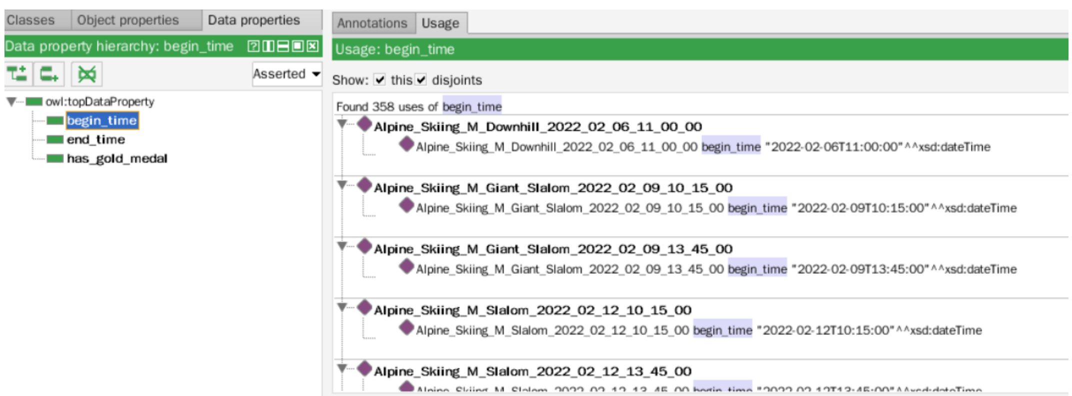


Figure 4: Data properties: begin, end time

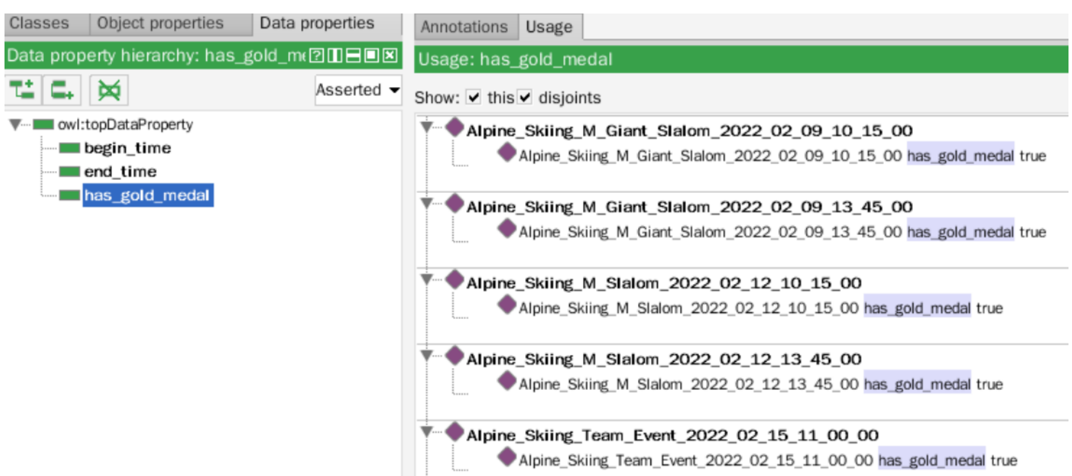


Figure 5: Data properties: gold medals