Reasoning in Description Logic

INFS4206/INFS7206

School of Information Technology and Electrical Engineering

Semester 2, 2006

THE UNIVERSITY OF QUEENSLAND
AUSTRALIA

What is logic?

- The art of reasoning
- The art of formal reasoning
- The art of formal reasoning about true/false statements
- The study of valid inference patterns. How to derive valid conclusions from given premises

Reasoning in Description Logic School of Information Technology and Electrical Engineering

Logic, logic and logic!

- Propositional logic
  - proposition and connective for propositions (propositions are true or false)
  - $p \wedge q$
- First-Order Logic
  - predicates, constants, variables, quantifiers to build propositions
  - $\forall x \exists y (P(x) \wedge Q(y))$
- Description Logic
  - Concepts, Roles, Concept constructors
  - $C \sqcap D$

Representing knowledge in Description Logics

A Knowledge Base (KB) in Description Logic consists of

TBox: Concepts definitions

- equivalence axioms $C \equiv D$

$$Course \equiv ITcourse \sqcap EEcourse$$

- inclusion axioms $C \sqsubseteq D$

$$Lecturer \sqsubseteq \exists teaches.Course$$

ABox: individual assertions

$$Lecturer(GUIDO)$$
$$takes(S123, INFS4206)$$
$$\forall teaches.ITcourse(BOB)$$
$$Course(COMP6801)$$

Your homework (Part 1)

Give suitable names to the concepts introduced as examples, and include them in a knowledge base

$$
\begin{aligned}
CommonCourses &\equiv ITcourse \sqcap EEcourse \\
ITstudent &\sqsubseteq Student \sqcap \forall takes.ITcourse \\
TeachingStaff &\sqsubseteq Lecturer \sqcap \exists teaches.\top \\
ITEEcourses &\equiv ITcourse \sqcup EEcourse \\
EElecturer &\sqsubseteq \exists teaches.EEcourse \\
BusyLecturer &\sqsubseteq\ \geq 2 teaches \\
BusyITEELecturer &\sqsubseteq\ \geq 2 teaches.(ITcourse \sqcap EEcourse) \\
IHaveNoIdea &\equiv \neg(Student \sqcap Lecturer) \\
GuidoCourseThisSemester &\equiv \{INFS4206, INFS1200, INFS4203\} \\
GuidoCourse &\equiv teaches : GUIDO
\end{aligned}
$$

Your homework (Part 2)

Write in Description Logic (you can use all constructors given in this lecture) the sentences of slides "Valuation of ∀ and ∃ (1) and (2)"

- There is a lecturer who teaches INFS4206

$$Lecturer \sqcap \exists teaches.\{INFS4206\}$$
$$Lecturer \sqcap teaches^{-} : INFS4206$$
$$\{INFS4206\} \sqsubseteq \exists teaches^{-}.Lecturer$$

- Guido teaches every course

$$Course \sqsubseteq teaches : GUIDO$$

- Bob teaches some courses

$$\{BOB\} \sqsubseteq \exists teaches.Course$$

Your homework (Part 2)

Write in Description Logic (you can use all constructors given in this lecture) the sentences of slides "Valuation of $\forall$ and $\exists$ (1) and (2)"

- Every lecturer teaches at least one course

$$Lecturer \sqsubseteq \exists teaches.Course$$

- Every student teaches at least one course

$$Student \sqsubseteq \exists teaches.Course$$

- There a student who teaches every course

???

Reasoning about TBoxes

Satisfiability A concept C is satisfiable wrt a TBox $\mathcal{T}$ iff (if and only if) there is a model $\mathcal{I}$ of $\mathcal{T}$ such that $C^{\mathcal{I}}$ is not empty.

Subsumption A concept C is subsumed by a concept D wrt $\mathcal{T}$ iff for every model $\mathcal{I}$ of $\mathcal{T}$ we have $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$. And we denote it by $\mathcal{T} \models C \sqsubseteq D$.

Equivalence Two concepts C and D are equivalent wrt $\mathcal{T}$ iff for for every model $\mathcal{I}$ of $\mathcal{T}$ we have $C^{\mathcal{I}} = D^{\mathcal{I}}$. And we denote it by $\mathcal{T} \models C \equiv D$.

Disjointness Two concepts C and D are disjoint wrt $\mathcal{T}$ iff for for every model $\mathcal{I}$ of $\mathcal{T}$ we have $C^{\mathcal{I}} \cap D^{\mathcal{I}} = \emptyset$.

Two kinds of knowledge

There are two kinds of derived knowledge

- knowledge/inferences depending on the definitions of the concepts in a TBox

$$\left\{ \begin{array}{c} Student \sqsubseteq \neg Lecturer, \\ Lecturer \sqsubseteq \exists teaches.Course \end{array} \right\} \models Student \sqsubseteq \neg \exists teaches.Course$$

- knowledge/inferences depending on the logical structure of the concepts

$$\models \exists teaches.ITcourse \sqsubseteq \exists teaches.(ITcourse \sqcup EEcourse)$$

Reduction to subsumption

Theorem

For concepts C and D we have

- C *is unsatisfiable iff* $\models C \sqsubseteq \bot$
- $\models C \equiv D$ *iff* $\models C \sqsubseteq D$ *and* $\models D \sqsubseteq C$
- C *and* D *are disjoint iff* $\models C \sqcap D \sqsubseteq \bot$

Reduction to unsatisfiability

Theorem

For concepts C and D we have

- $\models C \sqsubseteq D$ *iff* $C \sqcap \neg D$ *is unsatisfiable*
- $\models C \equiv D$ *iff both* $C \sqcap \neg D$ *and* $\neg C \sqcap D$ *are unsatisfiable*
- C *and* D *are disjoint iff* $C \sqcap D$ *is unsatisfiable*

Reducing unsatisfiability

Theorem

Let C be a concept. The following are equivalent

- C *is unsatisfiable*
- $\models C \sqsubseteq \bot$
- $\models C \equiv \bot$
- C *and* $\top$ *are disjoint*

Reasoning about ABoxes

An ABox $\mathcal{A}$ is consistent wrt to a TBox $\mathcal{T}$ iff there is an interpretation $\mathcal{I}$ which is a model of both $\mathcal{A}$ and $\mathcal{T}$.

Instance checking An assertion α is entailed by $\mathcal{A}$ ($\mathcal{A} \models \alpha$) iff every model $\mathcal{I}$ of $\mathcal{A}$ is also a model of α.

Equivalently $\mathcal{A} \models C(a)$ iff $\mathcal{A} \cup \{\neg C(a)\}$ is inconsistent

Theorem

A concept C is satisfiable iff $\{C(a)\}$ is consistent for an arbitrary a

Structural subsumption algorithm for $\sqcap$, $\forall R.C$, atomic negation and $\bot$

- Syntactical mechanism that works for simple description logics
- based on substitution of fundamental equivalences
  - $A \sqcap A \equiv A$
  - $A \sqcap B \equiv B \sqcap A$
  - $A \sqcap (B \sqcap C) \equiv (A \sqcap B) \sqcap C \equiv A \sqcap B \sqcap C$
  - $\forall R.(A \sqcap B) \equiv (\forall R.A) \sqcap (\forall R.B)$
  - $A \sqcap \neg A \equiv \bot$
  - $A \sqcap \bot \equiv \bot$

Normal Forms

- an atomic concept (A) is in normal form
- the negation of an atomic concept ($\neg A$) is in normal form
- $\bot$ is in normal form
- C is in normal form if has the following form

$$A_1 \sqcap \ldots \sqcap A_n \sqcap \forall R_1.C_1 \sqcap \ldots \forall R_m.C_m$$

where $A_1, \ldots, A_n$ are atomic concepts, $R_1, \ldots, R_m$ are distinct roles, and $C_1, \ldots, C_m$ are concepts in normal forms.

Theorem

Every concept C is equivalent to a concept C' in normal form.

Checking for subsumption

- $\bot$ is subsumed by every concept
- D subsumes C (or C is subsumed by D) if

$$D = A_1 \sqcap \ldots \sqcap A_n \sqcap \forall R_1.D_1 \sqcap \ldots \forall R_k.D_k$$

and

$$C = B_1 \sqcap \ldots \sqcap B_m \sqcap \forall S_1.C_1 \sqcap \ldots \forall S_l.C_l$$

and

  - for every i, $1 \leq i \leq n$ there is j, $1 \leq j \leq m$ such that $A_i = B_j$ and
  - for every i, $1 \leq i \leq k$ there is j, $1 \leq j \leq l$ such that $R_i = S_j$ and $C_j \sqsubseteq D_j$.

Examples

- $A \sqsubseteq A$
- $A \sqcap B \sqsubseteq A$
- $\forall R.(A \sqcap B) \sqsubseteq \forall R.A$
- $A \sqcap B \sqcap \forall R.(C \sqcap D \sqcap \forall S.E) \sqcap \forall T.\bot \sqsubseteq B \sqcap \forall T.F \sqcap \forall R.(C \sqcap \forall S.E)$

Readings

Franz Baader and Werner Nutt. *Basic Description Logics*. In Baader, Calvanese, McGuinnes Nardi and Patel-Schneider, (eds). **The Description Logics Handbook**, chapter 2, pages 43-95. Cambridge University Press, 2003. Sections 2.2.2.1, 2.2.2.2, 2.2.3, 2.2.4, 2.3.1