Représentation de connaissances : La logique de description

Yue Ma

Contact : yue.ma@u-psud.fr

Outline

Description Logics

subfield of knowledge representation

which is a subfield of artificial intelligence

Description

comes from concept description:

a formal expression that determines a set of individuals

Logics

comes from the semantics being defined using logic:

most Description Logics are fragments of First Order Logic

Knowledge Representation

Goal [Brachman & Nardi, 2003]

> develop formalisms for providing high-level description of the world that can be effectively used to build intelligent applications

- formalism: well-defined syntax; formal, unambiguous semantics
- high-level description: only relevant aspects represented
- intelligent applications: reason about the knowledge; infer implicit knowledge
- effectively used: practical reasoning tools and efficient implementations

Syntax

- explicit symbolic representation of the knowledge
- not implicit (as e.g. in neural networks)

$$
\begin{array}{ll}
\mathit{Woman} \equiv \mathit{Person} \sqcap \mathit{Female} & \\
\mathit{Man} \equiv \mathit{Person} \sqcap \neg \mathit{Female} & \mathit{Male}(\mathit{john}) \\
\mathit{Mother} \equiv \mathit{Woman} \sqcap \exists \mathit{hasChild}.\top & \mathit{Male}(\mathit{marc}) \\
 & \mathit{Male}(\mathit{stephen}) \\
 & \mathit{Male}(\mathit{jason}) \\
\mathit{hasChild}(\mathit{stephen}, \mathit{marc}) & \mathit{Female}(\mathit{jill}) \\
\mathit{hasChild}(\mathit{marc}, \mathit{anna}) & \mathit{Female}(\mathit{anna}) \\
\mathit{hasChild}(\mathit{john}, \mathit{maria}) & \mathit{Female}(\mathit{maria}) \\
\mathit{hasChild}(\mathit{anna}, \mathit{jason}) &
\end{array}
$$

Semantics

connection between the symbolic representation and the “real world” entities it represents

Declarative semantics

- map symbols to an abstraction of the “world” (interpretation)
- notion of when a symbolic expression is true in the world (model)

NO procedural semantics:

should not just express how specific programs should behave

Expressive Power

(what can be expressed) depends on syntax and semantics

Equilibrium

- not too low: can all the relevant knowledge be represented?
- not too high: are all the elements really necessary for the application?

Reasoning

deduce implicit knowledge from the explicit representation

$$\forall x.\forall y.(male(y) \land \exists z.(child(x,z) \land child(z,y))) \rightarrow grandson(x,y)$$

$child(john, mary)$
$child(mary, paul)$
$male(paul)$

grandson(john,paul)

Knowledge Representation Systems

- should provide inference tools to deduce implicit consequences
- answers should depend on semantics; not on the syntactic representation (same semantics must yield the same answer)

Source : inspiré du cours de Rafael Peñaloza (TU Dresden)

Reasoning Procedures

Decision Procedure

- sound: all positive answers are correct
- complete: all negative answers are correct
- terminating: always provides an answer in finite time

Efficient

- ideally, optimal w.r.t. worst-case complexity of the problem

Practical

- easy to implement and optimize
- behave well in applications

Examples of Formalisms

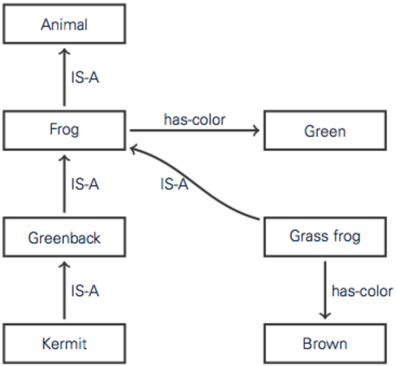
First-order logic

- satisfiability in FOL does not have a decision procedure
- is thus not an appropriate knowledge representation formalism

Propositional logic

- satisfiability in propositional logic has a decision procedure the problem is NP-complete
  - highly optimized SAT solver behave well in practice
  - however, expressive power is often insufficient

Terminological Knowledge



Terminological Knowledge

formalize the terminology (names) of the application domain:

- define important notions of the domain (classes, relations, objects)
- constrain the interpretations of these notions
- deduce consequences: subclass, instance relationships

Example (university domain)

- classes (concepts): Person, Teacher, Course, Student, . . .
- relations (roles): gives, attends, likes, . . .
- objects (individuals): DL WS13, Marcel, Daniel, . . .
- constraints:

  every course is given by a teacher,

  every student must attend at least one course

Ontologies

the modern name for knowledge bases

Applications

- semantic web enable a common understanding of notions for semantic labeling of Web content
- information retrieval support automatic extraction of information from text
- medicine formal definitions that can be used by doctors, patients, insurance companies, etc, to communicate with each other

Description Logics

a class of logic-based knowledge representation formalisms for representing terminological knowledge

Prehistory

early approaches for representing terminological knowledge

- semantic networks (Quillian, 1968)
- frames (Minsky, 1975)

problems with missing semantics led to

- structured inheritance networks
- the first DL system KL-ONE

History of Description Logics

- Phase 1

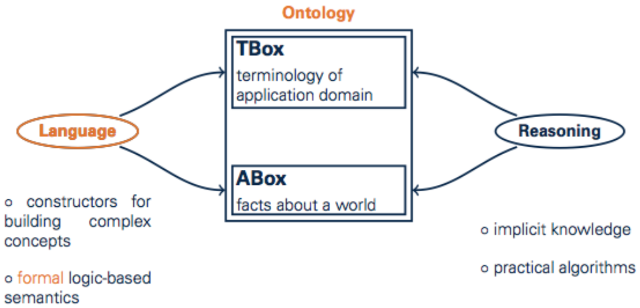
  implementation of systems based on incomplete structural subsumption algorithms

- Phase 2
  - tableau-based algorithms and complexity results
  - first tableau-based systems (Kris, Crack)
  - first formal study of optimization methods
- Phase 3
  - tableau-based algorithms for very expressive DLs
  - highly-optimized tableau-based systems (FaCT, Racer)
  - relationship to modal logic and FOL

History of Description Logics

- Phase 4
  - Web Ontology Language (OWL) based on DL
  - industrial-strength reasoners and ontology editors
  - light-weight (tractable) DLs
- Phase 5
  - non-standard reasoning
  - ontology management
  - semantic extensions

Structure of Description Logic Systems



 Source : inspiré du cours de Rafael Peñaloza (TU Dresden)

Outline

Syntax of $\mathcal{ALC}$

Let N_C and N_R two disjoint sets of concept names and role names, respectively.

$\mathcal{ALC}$ (complex) concepts are defined by induction:

- if $A \in N_C$,then A is an $\mathcal{ALC}$ concept
- if C, D are $\mathcal{ALC}$ concepts and $r \in N_R$, then the following are $\mathcal{ALC}$ concepts:
  - $C \sqcap D$ (conjunction)
  - $C \sqcup D$ (disjunction)
  - $\neg C$ (negation)
  - $\exists r.C$ (existential restriction)
  - $\forall r.C$ (value restriction)
- Abbreviations:
  - $\top = A \sqcup \neg A$ (top)
  - $\bot = A \sqcap \neg A$ (bottom)

Notations

- concept names are also called atomic concepts
- all other concepts are called complex
- instead of $\mathcal{ALC}$ concept, we often say concept

- A, B stand for concept names
- C, D for (complex) concepts
- r, s for role names

Try to write down your concepts.

Semantics of $\mathcal{ALC}$

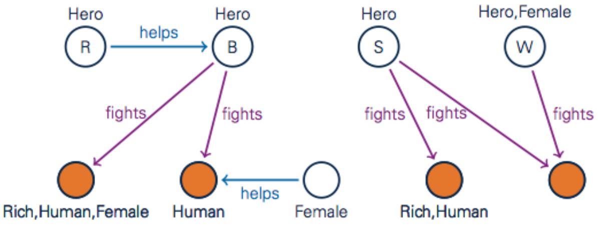
An interpretation $I = (\Delta^I, \cdot^I)$ consists of:

- a non-empty domain Δ^I , and
- an extension mapping $\cdot^I$ (also called interpretation function):
  - $A^I \subseteq \Delta^I$ for all $A \in N_C$ (concepts interpreted as sets)
  - $r^I \subseteq \Delta^I \times \Delta^I$ for all $r \in N_R$ (roles interpreted as binary relations)

The extension mapping is extended to concept descriptions as follows:

$$
\begin{array}{rcl}
(C \sqcap D)^I & := & C^I \cap D^I \\
(C \sqcup D)^I & := & C^I \cup D^I \\
(\neg C)^I & := & \Delta^I \setminus C^I \\
(\exists r.C)^I & := & \{d \in \Delta^I \mid \text{there is } e \in \Delta^I \text{ with } (d, e) \in r^I \text{ and } e \in C^I\} \\
(\forall r.C)^I & := & \{d \in \Delta^I \mid \text{for all } e \in \Delta^I \text{ with } (d, e) \in r^I, \text{ it holds } e \in C^I\}
\end{array}
$$

Interpretation Example



$$(Hero \sqcap \exists fights.Human)^I = \{B, S\}$$
$$(Hero \sqcap \forall fights.(Rich \sqcup \neg Human))^I = \{R, S, W\}$$
$$(\forall helps.Human)^I = \Delta^I \setminus \{R\}$$

DL vs FOL

- ALC can be seen as a fragment of First-order Logic
  - concept names are unary predicates
  - role names are binary predicates
- Interpretations can obviously be seen as first-order interpretations for this signature
- Concepts correspond to FOL formulae with one free variable

Translation to FOL

Syntactic translation $C \mapsto \pi_x(C)$

$$
\begin{aligned}
\pi_x(A) &:= A(x) \text{ for } A \in N_C \\
\pi_x(C \sqcap D) &:= \pi_x(C) \wedge \pi_x(D) \\
\pi_x(C \sqcup D) &:= \pi_x(C) \vee \pi_x(D) \\
\pi_x(\neg C) &:= \neg \pi_x(C) \\
\pi_x(\exists r.C) &:= \exists y.(r(x,y) \wedge \pi_y C), y \text{ new variable different from} x \\
\pi_x(\forall r.C) &:= \forall y.(r(x,y) \rightarrow \pi_y C)
\end{aligned}
$$

Example.

$\pi_x(\forall r.(A \sqcap \exists s.B)) = ?$

Translation to FOL

Syntactic translation $C \mapsto \pi_x(C)$

$$
\begin{aligned}
\pi_x(A) &:= A(x) \text{ for } A \in N_C \\
\pi_x(C \sqcap D) &:= \pi_x(C) \wedge \pi_x(D) \\
\pi_x(C \sqcup D) &:= \pi_x(C) \vee \pi_x(D) \\
\pi_x(\neg C) &:= \neg \pi_x(C) \\
\pi_x(\exists r.C) &:= \exists y.(r(x,y) \wedge \pi_y C), y \text{ new variable different from} x \\
\pi_x(\forall r.C) &:= \forall y.(r(x,y) \rightarrow \pi_y C)
\end{aligned}
$$

Example.
$\pi_x(\forall r.(A \sqcap \exists s.B)) = ?$

Lemma. C and $\pi_x(C)$ have the same extension; that is,

$$C^I = \{d \in \Delta^I \mid I \models \pi_x(C)(d)\}.$$

To define what is an ontology, we need the following defintions:

- TBox, ABox, GCIs, Concept definitions. Assertions.

GCIs (General Concept Inclusions) and TBoxes

Definitions.

- A general concept inclusion (GCI) is of the form $C \sqsubseteq D$, where C, D are concepts.
- A TBox $\mathcal{T}$ is a finite set of GCIs.
- An interpretation I satisfies the GCI $C \sqsubseteq D$ iff $C^I \subseteq D^I$. I is a model of a TBox T iff it satisfies all GCIs in T.
- Two TBoxes are equivalent if they have the same models.

Examples.

$Hero \sqcap Villain \sqsubseteq \bot$ (two concepts are disjoint)

$Cold \sqcap \exists causedBy.Virus \sqsubseteq Disease$

$Kitchen \sqcup Bathroom \sqsubseteq \exists hasWashbasin.\top$

Defining a Concept (aka. Concept Definitions)

Definitions.

- A concept definition is of the form $A = C$ where
  - A is a concept name.
  - C is a concept description.
- An interpretation I satisfies the concept definition $A = C$ if $A^I = C^I$.

Examples.

$Heroine = Hero \sqcap Female$

$MutantCriminal = Criminal \sqcap \exists fights.Mutant$

$Student = People \sqcap \exists regists.(School \sqcup University)$

Assertions and ABoxes

Definitions.

- An assertion is of the form $C(a)$ (concept assertion) or $r(a, b)$ (role assertion) where C is a concept, r a role, and a, b are individual names from a set N_I (disjoint with N_C, N_R)
- An ABox $\mathcal{A}$ is a finite set of assertions
- An interpretation I is a model of the ABox $\mathcal{A}$ if it satisfies all its assertions:
  - $a^I \in C^I$ for all $C(a) \in \mathcal{A}$.
  - $(a^I, b^I) \in R^I$ for all $R(a, b) \in \mathcal{A}$.

Examples.

$Kitchen(room1)$ (the room1 is a kitchen)

$locatedIn(whitechair1, room1)$ (the whitechair1 is in the room1)

$\neg Student(Jean)$ (Jean is not a student)

Ontology

Definitions.

- An ontology $O = (\mathcal{T}, \mathcal{A})$ consists of a TBox $\mathcal{T}$ and an ABox $\mathcal{A}$.
- The interpretation I is a model of the ontology $O = (\mathcal{T}, \mathcal{A})$ iff it is a model of $\mathcal{T}$ and a model of $\mathcal{A}$.

Examples.
Many ontologies from http://swoogle.umbc.edu and http://bioportal.bioontology.org

Let’s define the interesting reasoning services that we can benefit from ontologies

Terminological Reasoning of an Ontology

Let $\mathcal{T}$ be a TBox. Terminological reasoning refers to deciding the following problems:

- Satisfiability: a concept C is satisfiable w.r.t. $\mathcal{T}$ if and only if there is a model I of $\mathcal{T}$ such that $C^I \neq \emptyset$.
- Subsumption: C is subsumed by D w.r.t. $\mathcal{T}$ if and only if $C^I \subseteq D^I$ for all models I of $\mathcal{T}$.
- Equivalence: C is equivalent to D w.r.t. $\mathcal{T}$ if and only if $C^I = D^I$ for all models I of $\mathcal{T}$.
- Entailment: An ontology O entails $C \sqsubseteq D$, written $O \models C \sqsubseteq D$ if and only if $C^I \subseteq D^I$ (resp. $a^I \in C^I$, $(a^I, b^I) \in R^I$) for all models I of O.

If $\mathcal{T} = \emptyset$, we simply remove the âw.r.t T â from the names.

Examples (next page)

Examples.

- $A \sqcap \neg A$ is unsatisfiable
- $\forall r.A \sqcap \forall r.\neg A$ is unsatisfiable
- $\forall r.A \sqcap \exists r.\neg A$ is satisfiable
- $\exists r.(A \sqcap B)$ is subsumed by $\exists r.A$
- $\{A \sqsubseteq B, B \sqsubseteq C\} \models A \sqsubseteq C$ (entailment)

Assertional Reasoning of an Ontology

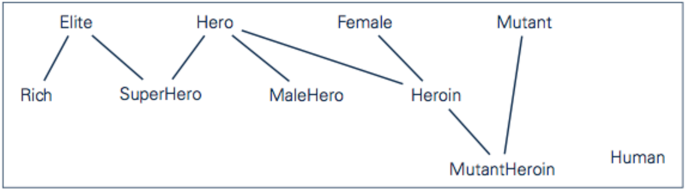
Let $O = (\mathcal{T}, \mathcal{A})$ be an ontology with TBox $\mathcal{T}$ and ABox $\mathcal{A}$. Assertional reasoning refers to deciding the following problems:

- Consistency: O is consistent iff O has a model.
- Instance: a is an instance of a concept C w.r.t O iff $a^I \in C^I$ for all models I of O

TBox Classification

Computing the subsumption relations between all concept names in $\mathcal{T}$:

$$
\begin{aligned}
\text{Heroine} &\equiv \text{Hero} \sqcap \text{Female} \\
\text{MaleHero} &\equiv \text{Hero} \sqcap \neg\text{Female} \\
\text{MutantHeroine} &\equiv \text{Heroine} \sqcap \text{Mutant} \\
\text{Elite} &\equiv \text{Rich} \sqcup \neg\text{Human} \\
\text{Superhero} &\equiv \text{Hero} \sqcap \text{Elite}
\end{aligned}
$$



Source : inspiré du cours de Rafael Peñaloza (TU Dresden)

Realization

Computing the most specific concept names to which an individual belongs

$$\begin{aligned}
\text{Heroine} &\equiv \text{Hero} \sqcap \text{Female} \\
\text{MaleHero} &\equiv \text{Hero} \sqcap \neg\text{Female} \\
\text{MutantHeroine} &\equiv \text{Heroine} \sqcap \text{Mutant} \\
\text{Elite} &\equiv \text{Rich} \sqcup \neg\text{Human} \\
\text{Superhero} &\equiv \text{Hero} \sqcap \text{Elite}
\end{aligned}$$

Hero(Superman)

Superman is an instance of

Hero, MaleHero, Elite, Superhero