Chapter 15:
Logical Representations of Sentence Meaning

Miyamae Yuichi

Table of Contents

Table of Contents

Introduction of Meaning Representations

- meaning representations (MR): Formal structures to capture the meaning of linguistic expressions

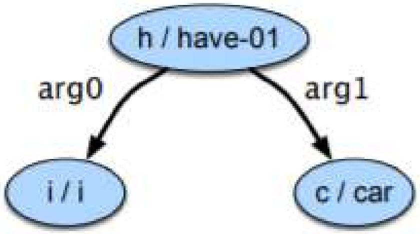
- Tasks below require representations that link the linguistic elements to the knowledge of the world
  - Deciding what to order at a restaurant by reading a menu
  - Giving advice about where to go to dinner
  - Following a recipe
  - Learning to use a new software by reading the manual
  - Giving advice on using software

Introduction of Meaning Representations

- MRs for *I have a car*

$$\exists e,y\ Having(e) \wedge Haver(e,Speaker) \wedge HadThing(e,y) \wedge Car(y)$$

First-Order Logic




```
(h / have-01
       arg0: (i / i)
       arg1: (c / car))
```

```
Having:
       Haver:   Speaker
       HadThing:   Car
```

Abstract Meaning Representation

framed-based or slot-filler representation

Introduction of Meaning Representations

- They can be viewed from 2 perspectives
  - As representations of the meaning of the particular linguistic inputs *I have a car*
  - As representations of the state of affairs in some world

- semantic parsing / semantic analysis: process to create representations and assign to linguistic inputs

- computational semantics: designing meaning representations + associated semantic parsers

Table of Contents

1. Desiderata for Representations

Consider a system that gives restaurant advice to tourists based on a knowledge base (KB).

1. **Verifiability**

- The system can compare the state of affairs described by MR to ones in some world as models in a KB

Does Maharani serve vegetarian food?

$\Rightarrow$ $Serves(Maharani, VegetarianFood)$

then match it against KB and find a representation matching it

1. Desiderata for Representations

2. **Unambiguous Representations**

- MRs cannot be ambiguous

- A single linguistic expression can have 2 meanings

  I wanna eat someplace that's close to ICSI.

  - The speaker wants to eat at some nearby location
  - The speaker wants to devour some nearby location

- vagueness doesn't give rise to multiple representations

  I want to eat Italian food.

1. Desiderata for Representations

3. **Canonical Form**

- Distinct inputs that mean the same thing should have the same meaning representation

Does Maharani serve vegetarian food?

- *Does Maharani have vegetarian dishes?*
- *Does Maharani serve vegetarian fare?*
- *Do they have vegetarian food at Maharani?*
- *Are vegetarian dishes served at Maharani?*

1. Desiderata for Representations

4. **Inference and Variables**

- inference: The system need to draw valid conclusions based on the MR of inputs and its background knowledge

Can vegetarians eat at Maharani?

- The same answer as *Does Maharani serve vegetarian food?*
- But it needs a connection between what vegetarians eat and what vegetarian restaurants serve

1. Desiderata for Representations

4. **Inference and Variables**

- variables: Meaning representations can handle indefinite references

I'd like to find a restaurant where I can get vegetarian food.

⇨ $Serves(x, VegetarianFood)$

1. Desiderata for Representations

5. **Expressiveness**

- Meaning representation must be expressive enough to handle a wide range of subject matter

- First-Order Logic is expressive enough (explained later)

Table of Contents

2. Model-Theoretic Semantics

- model: formal construct that stands for the particular state of affairs in the world
  - Expressions in MR can be mapped to elements of the model

- Vocabulary of MR
  - non-logical vocabulary:
    objects, properties of objects, relations among objects
  - logical vocabulary:
    symbols, operators, quantifiers ($\forall$, $\exists$), links, etc.

2. Model-Theoretic Semantics

Domain	$\mathscr{D} = \{a, b, c, d, e, f, g, h, i, j\}$
Matthew, Franco, Katie and Caroline	a, b, c, d
Frasca, Med, Rio	e, f, g
Italian, Mexican, Eclectic	h, i, j
Properties	
Noisy	$Noisy = \{e, f, g\}$
Frasca, Med, and Rio are noisy	
Relations	
Likes	$Likes = \{\langle a, f\rangle, \langle c, f\rangle, \langle c, g\rangle, \langle b, e\rangle, \langle d, f\rangle, \langle d, g\rangle\}$
Matthew likes the Med	
Katie likes the Med and Rio	
Franco likes Frasca	
Caroline likes the Med and Rio	
Serves	$Serves = \{\langle f, j\rangle, \langle g, i\rangle, \langle e, h\rangle\}$
Med serves eclectic	
Rio serves Mexican	
Frasca serves Italian	

Figure 15.2 A model of the restaurant world.

2. Model-Theoretic Semantics

Domain	$\mathscr{D} = \{a,b,c,d,e,f,g,h,i,j\}$
Matthew, Franco, Katie and Caroline	a,b,c,d
Frasca, Med, Rio	e,f,g
Italian, Mexican, Eclectic	h,i,j
Properties	
Noisy	$Noisy = \{e,f,g\}$
Frasca, Med, and Rio are noisy	
Relations	
Likes	$Likes = \{\langle a,f\rangle,\langle c,f\rangle,\langle c,g\rangle,\langle b,e\rangle,\langle d,f\rangle,\langle d,g\rangle\}$
Matthew likes the Med	
Katie likes the Med and Rio	
Franco likes Frasca	
Caroline likes the Med and Rio	

extensional

non-logical vocabulary $\xrightarrow{\text{interpretation}}$ denotation

2. Model-Theoretic Semantics

Relations

Likes $\qquad$ $Likes = \{\langle a,f\rangle, \langle c,f\rangle, \langle c,g\rangle, \langle b,e\rangle, \langle d,f\rangle, \langle d,g\rangle\}$

Matthew likes the Med

Katie likes the Med and Rio

Franco likes Frasca

Caroline likes the Med and Rio

Matthew likes Frasca → $Likes(a, e)$ → false

Katie likes the Rio and Matthew likes the Med.

- *Katie likes the Rio* → $Likes(c, g)$
- *Matthew likes the Med* → $Likes(a, f)$
- truth-conditional semantics: method for determining the truth of a complex expression

Table of Contents

3. First-Order Logic (FOL)

$$
\begin{array}{rcl}
\textit{Formula} & \rightarrow & \textit{AtomicFormula} \\
& | & \textit{Formula Connective Formula} \\
& | & \textit{Quantifier Variable}, \ldots \textit{ Formula} \\
& | & \neg \textit{ Formula} \\
& | & (\textit{Formula}) \\
\textit{AtomicFormula} & \rightarrow & \textit{Predicate}(\textit{Term}, \ldots) \\
\textit{Term} & \rightarrow & \textit{Function}(\textit{Term}, \ldots) \\
& | & \textit{Constant} \\
& | & \textit{Variable} \\
\textit{Connective} & \rightarrow & \wedge \mid \vee \mid \Longrightarrow \\
\textit{Quantifier} & \rightarrow & \forall \mid \exists \\
\textit{Constant} & \rightarrow & A \mid \textit{VegetarianFood} \mid \textit{Maharani} \cdots \\
\textit{Variable} & \rightarrow & x \mid y \mid \cdots \\
\textit{Predicate} & \rightarrow & \textit{Serves} \mid \textit{Near} \mid \cdots \\
\textit{Function} & \rightarrow & \textit{LocationOf} \mid \textit{CuisineOf} \mid \cdots
\end{array}
$$

Figure 15.3 A context-free grammar specification of the syntax of First-Order Logic representations. Adapted from Russell and Norvig (2002).

3-1. Basic Elements of FOL

- term: device for referring objects
  - constant: refer to specific objects

    e.g. A, B, Maharani, Harry
  - function: correspond to genitives

    e.g. $LocationOf(Frasca)$ = Frasca's location
  - variable: refer to anonymous objects

    e.g. x, y

- predicate: symbol that refer to the relations

    e.g. $Serves(Maharani,\ VegetarianFood)$

3-1. Basic Elements of FOL

Maharani serves vegetarian food.

$$Serves(Maharani, VegetarianFood)$$

Maharani is a restaurant.

$$Restaurant(Maharani)$$

I only have five dollars and I don't have a lot of time.

$$Have(Speaker, FiveDollars) \wedge \neg Have(Speaker, LotOfTime)$$

logical connectives

3-2. Variables and Quantifiers

- variables: used in two ways through quantifiers
  - refer to particular anonymous objects
  - refer generically to all objects

- quantifiers:
  - $\exists$ (existential quantifier): there exists
  - $\forall$ (universal quantifier) : for all

3-2. Variables and Quantifiers

a restaurant that serves Mexican food near ICSI.

$$\exists x\ Restaurant(x) \wedge Serves(x, MexicanFood) \wedge Near((LocationOf(x), LocationOf(ICSI))$$

All vegetarian restaurants serve vegetarian food.

$$\forall x\ VegetarianRestaurant(x) \Rightarrow Serves(x, VegetarianFood)$$

3-3. Lambda Notation

- lambda notation: way to abstract from FOL.

  $\lambda x. P(x)$

- λ-reduction: apply to logical term to yield to new FOL

  $\lambda x. P(x)(A) \Rightarrow P(A)$

- currying: way of converting a predicate with multiple arguments into a sequence of single-argument predicates

  $\lambda x. \quad y. Near(x, y)$

  $\lambda x. \quad y. Near(x, y)(Bacaro) \Rightarrow \lambda y. Near(Bacaro, y)$

  $\lambda y. Near(Bacaro, y)(Centro) \Rightarrow Near(Bacaro, Centro)$

3-4. The Semantics of FOL

- FOL employs the model-theoretic approach to let non-logical vocabularies acquire their meanings

- The interpretation of formulas involving logical connectives is based on the truth table

P	Q	$\neg P$	$P \wedge Q$	$P \vee Q$	$P \implies Q$
False	*False*	*True*	*False*	*False*	*True*
False	*True*	*True*	*False*	*True*	*True*
True	*False*	*False*	*False*	*True*	*False*
True	*True*	*False*	*True*	*True*	*True*

Figure 15.4 Truth table giving the semantics of the various logical connectives.

3-5. Inference

- modus ponens: a form of inference by if-then reasoning

$$\frac{\begin{array}{l}\alpha \\ \alpha \Longrightarrow \beta\end{array}}{\beta}$$

If the antecedent (α) is true,
then the consequent (β) can be inferred

- Example of using modus ponens:

$$\frac{\begin{array}{l}VegetarianRestaurant(Leaf) \\ \forall x VegetarianRestaurant(x) \Longrightarrow Serves(x, VegetarianFood)\end{array}}{Serves(Leaf, VegetarianFood)}$$

3-5. Inference

- forward chaining:
  - If $\alpha \Rightarrow \beta$ and α, then β
  - A new fast is added to the KB
    → All applicable rules are found and applied
    → Results (new facts) are added to the KB
    → Repeat until no further facts are deduced

$$\frac{\begin{array}{l}\alpha \\ \alpha \implies \beta\end{array}}{\beta}$$

- backward chaining:
  - See if β is in the KB
    → If not, search for applicable rules in the KB
    → See if α is true

3-5. Inference

- abduction: plausible reasoning from β to α
  - if $\alpha \Rightarrow \beta$ and β, then α is plausible

$$\frac{\begin{array}{c}\alpha \\ \alpha \Longrightarrow \beta\end{array}}{\beta}$$

- Neither forward and backward chaining are complete
  - There are valid inferences that cannot be found by them
- resolution is sound and complete
  - but far more computationally expensive

Table of Contents

4. Event and State Representations

- state: conditions that remain unchanged
- event: changes in some state of affairs

- Predicates in FOL have fixed arity

$$Serves(Leaf, VegetarianFare)$$

- Choosing the number of arguments for *eat* is tricky
  - I ate.
  - I ate a turkey sandwich.
  - I ate a turkey sandwich for lunch at my desk.

4. Event and State Representations

- neo-Davidsonian event representations:
  - Donald Davidson introduced the notion of an event variable
  - e: event variable

I ate a turkey sandwich for lunch at my desk.

$$\begin{aligned}\exists e\ Eating(e) &\wedge Eater(e, Speaker)\\ &\wedge Eaten(e, TurkeySandwich)\\ &\wedge Meal(e, Lunch)\\ &\wedge Location(e, Desk)\end{aligned}$$

4-1. Representing Time

- temporal logic: the representation of time information
- tense logic: the ways verb tenses convey temporal info

- Sentences below have the same kind of event, but differ in verb tense
  - I arrived in New York.
  - I am arriving in New York.
  - I will arrive in New York.

$$\exists e\ Arriving(e) \wedge Arriver(e, Speaker) \wedge Destination(e, NewYork)$$

4-1. Representing Time

I arrived in New York.

$$\exists e, i, n\ Arriving(e) \wedge Arriver(e, Speaker) \wedge Destination(e, NewYork) \\ \wedge IntervalOf(e, i) \wedge EndPoint(i, n) \wedge Precedes(n, Now)$$

I am arriving in New York.

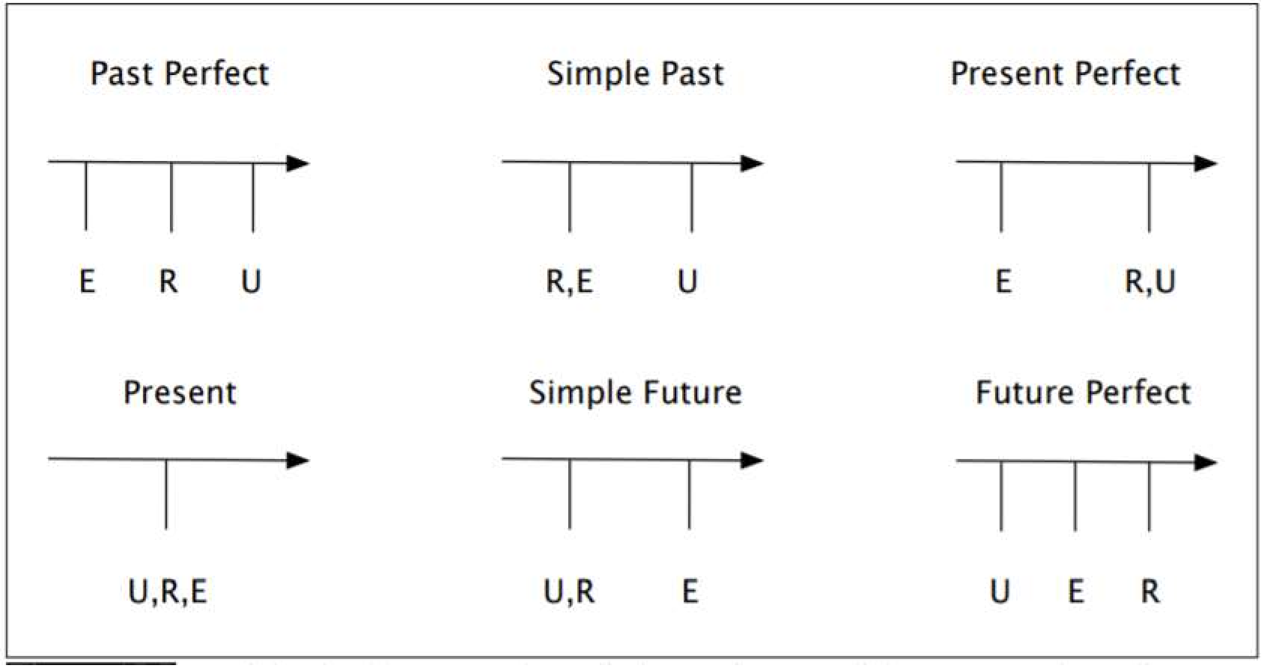
$$\exists e, i, n\ Arriving(e) \wedge Arriver(e, Speaker) \wedge Destination(e, NewYork) \\ \wedge IntervalOf(e, i) \wedge MemberOf(i, Now)$$

I will arrive in New York.

$$\exists e, i, n\ Arriving(e) \wedge Arriver(e, Speaker) \wedge Destination(e, NewYork) \\ \wedge IntervalOf(e, i) \wedge EndPoint(i, n) \wedge Precedes(Now, n)$$

4-1. Representing Time

- Reichenbach introduced the notion reference point
  - When Mary’s flight departed, I ate lunch.
  - When Mary’s flight departed, I had eaten lunch.



U: utterance
E: event
R: reference

4-2. Aspect

- stative: at a single point in time
  I know my departure gate.
- activity: no particular end point, over some span of time
  John is flying
- accomplishment:
  Sally booked her flight
- achievement:
  She found her gate

- telic eventualities: accomplishment + achievement

Table of Contents

5. Description Logics (DL)

- DL: useful and computationally tractable subsets of FOL

- terminology: set of categories, or concepts

- knowledge base = TBox + ABox
  - TBox contains the terminology
  - ABox contains facts about individuals

- ontology: hierarchical organization that the terminology is arranged into

5. Description Logics (DL)

- domain concepts

  FOL: $Restaurant(x)$ DL: Restaurant

- the fact about a domain element
  - FOL: $Restaurant(Frasca)$ DL: Restaurant(Frasca)

5. Description Logics (DL)

- There are 2 ways to arrange categories into a hierarchical structure
  1. Directly assert relations between categories that are related hierarchically
  2. Provide complete definitions for concepts and then rely on inference to provide hierarchical relationships

5. Description Logics (DL)

- subsumption relations: to specify a hierarchical structure

$$
\begin{aligned}
\mathsf{Restaurant} &\sqsubseteq \mathsf{CommercialEstablishment} \\
\mathsf{ItalianRestaurant} &\sqsubseteq \mathsf{Restaurant} \\
\mathsf{ChineseRestaurant} &\sqsubseteq \mathsf{Restaurant} \\
\mathsf{MexicanRestaurant} &\sqsubseteq \mathsf{Restaurant}
\end{aligned}
$$

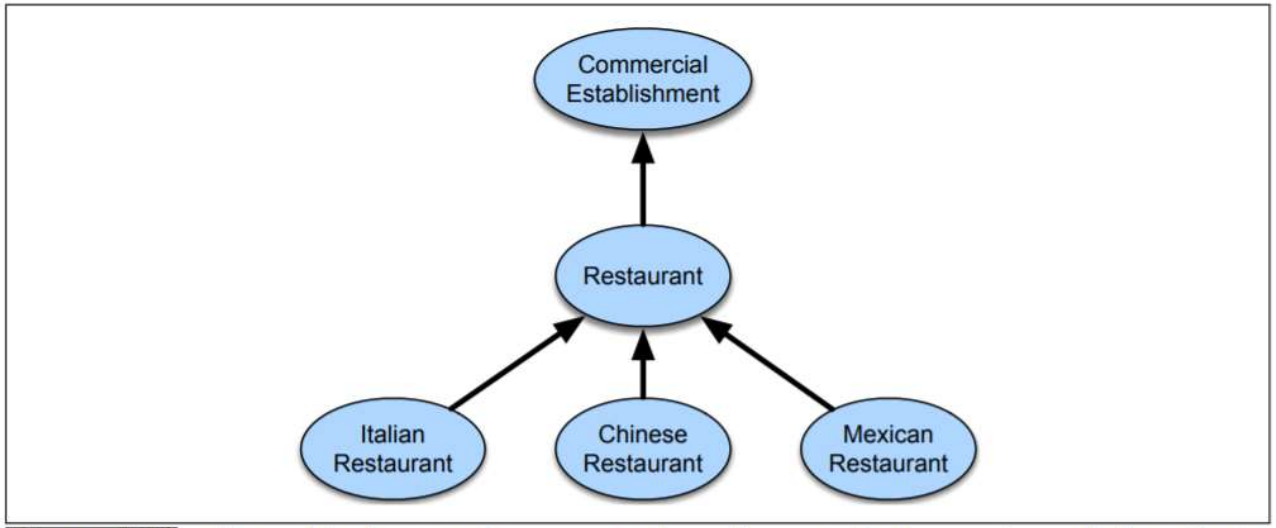


Figure 15.6 A graphical network representation of a set of subsumption relations in the restaurant domain.

5. Description Logics (DL)

- Chinese restaurants can't also be Italian restaurants.

$$\text{ChineseRestaurant} \sqsubseteq \textbf{not}\ \text{ItalianRestaurant}$$

- To specify that a set of subconcepts covers a category

$$\text{Restaurant} \sqsubseteq \\ (\textbf{or}\ \text{ItalianRestaurant ChineseRestaurant MexicanRestaurant})$$

5. Description Logics (DL)

- We don’t know anything about what makes a restaurant a restaurant

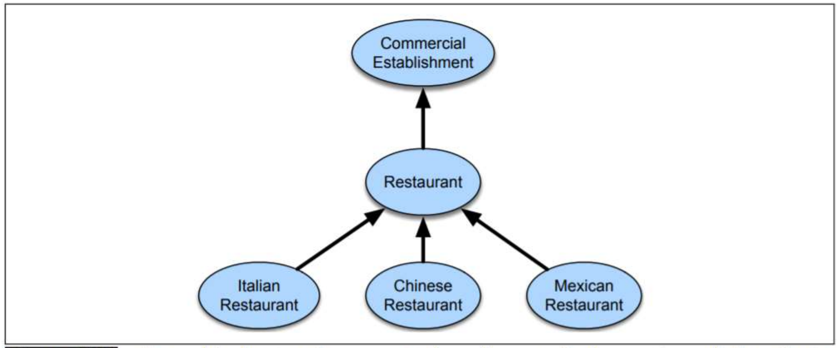


Figure 15.6 A graphical network representation of a set of subsumption relations in the restaurant domain.

- Needs for what it means to be a member of any of these categories

- In DL, such statements come in the form of relations between the concepts

5. Description Logics (DL)

- hasCuisine relation: what kinds of food restaurants serve
- hasPriceRange relation: how pricey restaurants tend to be

$$\begin{aligned}
\mathsf{MexicanCuisine} &\sqsubseteq \mathsf{Cuisine} \\
\mathsf{ItalianCuisine} &\sqsubseteq \mathsf{Cuisine} \\
\mathsf{ChineseCuisine} &\sqsubseteq \mathsf{Cuisine} \\
\mathsf{VegetarianCuisine} &\sqsubseteq \mathsf{Cuisine}
\end{aligned}$$

$$\begin{aligned}
\mathsf{ExpensiveRestaurant} &\sqsubseteq \mathsf{Restaurant} \\
\mathsf{ModerateRestaurant} &\sqsubseteq \mathsf{Restaurant} \\
\mathsf{CheapRestaurant} &\sqsubseteq \mathsf{Restaurant}
\end{aligned}$$

5. Description Logics (DL)

- Italian restaurants serve Italian cuisine.

$$\mathsf{ItalianRestaurant} \sqsubseteq \mathsf{Restaurant} \sqcap \exists \mathsf{hasCuisine.ItalianCuisine}$$

- When translated into FOL:

$$\begin{aligned} \forall x ItalianRestaurant(x) \;&\rightarrow\; Restaurant(x) \\ &\wedge (\exists y Serves(x,y) \wedge ItalianCuisine(y)) \end{aligned}$$

5. Description Logics (DL)

$$
\begin{aligned}
\mathsf{ItalianRestaurant} &\equiv \mathsf{Restaurant} \sqcap \exists \mathsf{hasCuisine.ItalianCuisine} \\
\mathsf{ModerateRestaurant} &\equiv \mathsf{Restaurant} \sqcap \mathsf{hasPriceRange.ModeratePrices}
\end{aligned}
$$

$$
\begin{aligned}
\mathsf{VegetarianRestaurant} \equiv\; & \mathsf{Restaurant} \\
& \sqcap \exists \mathsf{hasCuisine.VegetarianCuisine} \\
& \sqcap \forall \mathsf{hasCuisine.VegetarianCuisine}
\end{aligned}
$$

5. Description Logics (DL)

Inference

- subsumption: determine whether a superset/subset relationship exists between two concepts

$$\text{IlFornaio} \sqsubseteq \text{ModerateRestaurant} \sqcap \exists \text{hasCuisine.ItalianCuisine}$$

- What if we pose the following question to the system

$$\text{IlFornaio} \sqsubseteq \text{ItalianRestaurant}$$

$$\text{IlFornaio} \sqsubseteq \text{VegetarianRestaurant}$$

5. Description Logics (DL)

Inference

- implied hierarchy: a related reasoning task, repeated application of the subsumption operator

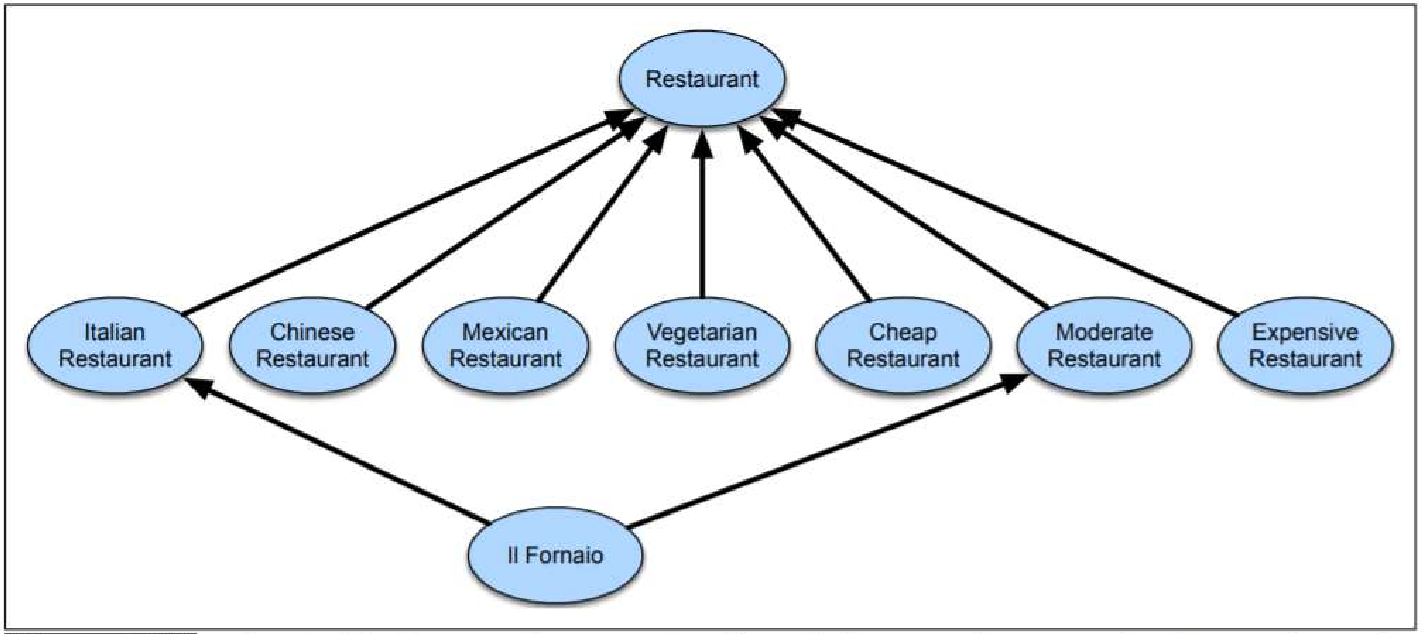


Figure 15.7 A graphical network representation of the complete set of subsumption relations in the restaurant domain given the current set of assertions in the TBox.

5. Description Logics (DL)

Inference

- instance checking: ask if an individual can be a member of a particular category

$$\mathsf{Restaurant(Gondolier)}$$

$$\mathsf{hasCuisine(Gondolier, ItalianCuisine)}$$

5. Description Logics (DL)

OWL and the Semantic Web

- Web Ontology Language (OWL)
  - OWL embodies a DL
  - This is one of the key components of Semantic Web

Table of Contents

6. Summary

- A major approach to meaning in computational linguistics involves the creation of **formal meaning representations** that capture the meaning-related content of linguistic inputs. These representations are intended to bridge the gap from language to common-sense knowledge of the world.
- The frameworks that specify the syntax and semantics of these representations are called **meaning representation languages**. A wide variety of such languages are used in natural language processing and artificial intelligence.
- Such representations need to be able to support the practical computational requirements of semantic processing. Among these are the need to determine **the truth of propositions**, to support **unambiguous representations**, to represent **variables**, to support **inference**, and to be sufficiently **expressive**.
- Human languages have a wide variety of features that are used to convey meaning. Among the most important of these is the ability to convey a **predicate-argument structure**.

6. Summary

- **First-Order Logic** is a well-understood, computationally tractable meaning representation language that offers much of what is needed in a meaning representation language.
- Important elements of semantic representation including **states** and **events** can be captured in FOL.
- **Semantic networks** and **frames** can be captured within the FOL framework.
- Modern **Description Logics** consist of useful and computationally tractable subsets of full First-Order Logic. The most prominent use of a description logic is the **Web Ontology Language** (OWL), used in the specification of the Semantic Web.