Lecture 12 of 42

Intro to First-Order Logic: Syntax and Semantics

William H. Hsu
Department of Computing and Information Sciences, KSU

KSOL course page: http://snipurl.com/v9v3
Course web site: http://www.kddresearch.org/Courses/CIS730
Instructor home page: http://www.cis.ksu.edu/~bhsu

Reading for Next Class:

Section 8.3 – 8.4, p. 253 - 266, Russell & Norvig 2nd edition
Handout, Nilsson & Genesereth, *Logical Foundations of Artificial Intelligence*

Lecture Outline

- **Reading for Next Class: 8.3-8.4 (p. 253-266), 9.1 (p. 272-274), R&N 2e**
- **Last Class: Propositional Logic, Sections 7.5-7.7 (p. 211-232), R&N 2e**
  - **Properties of sentences (and sets of sentences, *aka* knowledge bases)**
    - **entailment**
    - **provability/derivability**
    - **validity: truth in all models (*aka* tautological truth)**
    - **satisfiability: truth in some models**
  - **Properties of proof rules**
    - **soundness: $KB \vdash_i \alpha \Rightarrow KB \models \alpha$ (can prove only true sentences)**
    - **completeness: $KB \models \alpha \Rightarrow KB \vdash_i \alpha$ (can prove all true sentences)**
- **Still to Cover in Chapter 7: Resolution, Conjunctive Normal Form (CNF)**
- **Today: Intro to First-Order Logic, Sections 8.1-8.2 (p. 240-253), R&N 2e**
  - **Elements of logic: ontology and epistemology**
  - **Resolution theorem proving**
  - **First-order predicate calculus (FOPC) *aka* first order logic (FOL)**
- **Coming Week: Propositional and First-Order Logic (Ch. 8 – 9)**

Chapter 7 Concluded

- ◇ Knowledge-based agents
- ◇ Wumpus world
- ◇ Logic in general—models and entailment
- ◇ Propositional (Boolean) logic
- ◇ Equivalence, validity, satisfiability
- ◇ Inference rules and theorem proving
  - forward chaining
  - backward chaining
  - resolution

Inference: Review

$KB \vdash_i \alpha$ = sentence α can be derived from KB by procedure i

Consequences of KB are a haystack; α is a needle.
Entailment = needle in haystack; inference = finding it

Soundness: i is sound if
whenever $KB \vdash_i \alpha$, it is also true that $KB \models \alpha$

Completeness: i is complete if
whenever $KB \models \alpha$, it is also true that $KB \vdash_i \alpha$

Preview: we will define a logic (first-order logic) which is expressive enough to say almost anything of interest, and for which there exists a sound and complete inference procedure.

That is, the procedure will answer any question whose answer follows from what is known by the KB.

Validity and Satisfiability: Review

A sentence is valid if it is true in **all** models,

e.g., $True$, $A \lor \neg A$, $A \Rightarrow A$, $(A \land (A \Rightarrow B)) \Rightarrow B$

Validity is connected to inference via the Deduction Theorem:

$KB \models \alpha$ if and only if $(KB \Rightarrow \alpha)$ is valid

A sentence is satisfiable if it is true in **some** model

e.g., $A \lor B$, C

A sentence is unsatisfiable if it is true in **no** models

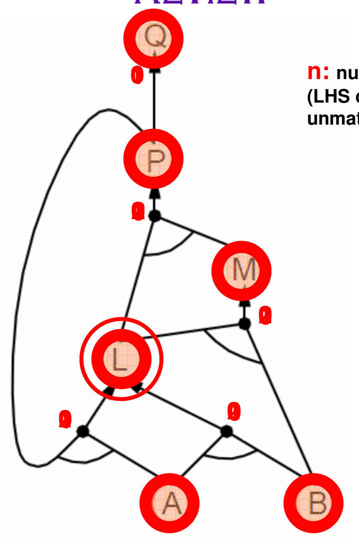
e.g., $A \land \neg A$

Satisfiability is connected to inference via the following:

$KB \models \alpha$ if and only if $(KB \land \neg\alpha)$ is unsatisfiable

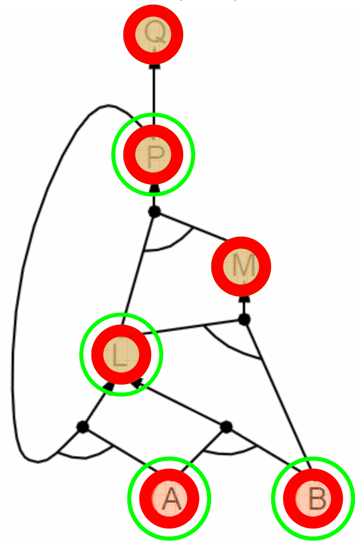
i.e., prove α by *reductio ad absurdum*

Forward Chaining Example: Review



n: number of antecedents (LHS conjuncts) still unmatched

Backward Chaining Example: Review



Forward *vs.* Backward Chaining: Review

FC is data-driven, cf. automatic, unconscious processing,
e.g., object recognition, routine decisions

May do lots of work that is irrelevant to the goal

BC is goal-driven, appropriate for problem-solving,
e.g., Where are my keys? How do I get into a PhD program?

Complexity of BC can be much less than linear in size of KB

Resolution [1]: Propositional Sequent Rule

Conjunctive Normal Form (CNF—universal)

$\text{conjunction of } \underbrace{\text{disjunctions of literals}}_{\text{clauses}}$

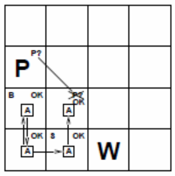
E.g., $(A \vee \neg B) \wedge (B \vee \neg C \vee \neg D)$

Resolution inference rule (for CNF): complete for propositional logic

$$\frac{\ell_1 \vee \cdots \vee \ell_k, \qquad m_1 \vee \cdots \vee m_n}{\ell_1 \vee \cdots \vee \ell_{i-1} \vee \ell_{i+1} \vee \cdots \vee \ell_k \vee m_1 \vee \cdots \vee m_{j-1} \vee m_{j+1} \vee \cdots \vee m_n}$$

where ℓ_i and m_j are complementary literals. E.g.,

$$\frac{P_{1,3} \vee P_{2,2}, \qquad \neg P_{2,2}}{P_{1,3}}$$



Resolution is sound and complete for propositional logic

Based on slide © 2004 S. Russell & P. Norvig. Reused with permission.

Resolution [2]: Conversion to Conjunctive Normal Form (CNF)

$B_{1,1} \Leftrightarrow (P_{1,2} \vee P_{2,1})$

1. Eliminate $\Leftrightarrow$, replacing $\alpha \Leftrightarrow \beta$ with $(\alpha \Rightarrow \beta) \wedge (\beta \Rightarrow \alpha)$.

   $(B_{1,1} \Rightarrow (P_{1,2} \vee P_{2,1})) \wedge ((P_{1,2} \vee P_{2,1}) \Rightarrow B_{1,1})$

2. Eliminate $\Rightarrow$, replacing $\alpha \Rightarrow \beta$ with $\neg\alpha \vee \beta$.

   $(\neg B_{1,1} \vee P_{1,2} \vee P_{2,1}) \wedge (\neg(P_{1,2} \vee P_{2,1}) \vee B_{1,1})$

3. Move $\neg$ inwards using de Morgan's rules and double-negation:

   $(\neg B_{1,1} \vee P_{1,2} \vee P_{2,1}) \wedge ((\neg P_{1,2} \wedge \neg P_{2,1}) \vee B_{1,1})$

4. Apply distributivity law ($\vee$ over $\wedge$) and flatten:

   $(\neg B_{1,1} \vee P_{1,2} \vee P_{2,1}) \wedge (\neg P_{1,2} \vee B_{1,1}) \wedge (\neg P_{2,1} \vee B_{1,1})$

Based on slide © 2004 S. Russell & P. Norvig. Reused with permission.

Resolution [3]: Algorithm

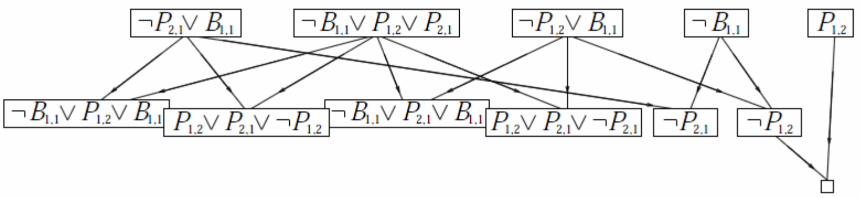
Proof by contradiction, i.e., show $KB \wedge \neg\alpha$ unsatisfiable

```
function PL-RESOLUTION(KB, α) returns true or false
   inputs: KB, the knowledge base, a sentence in propositional logic
           α, the query, a sentence in propositional logic

   clauses ← the set of clauses in the CNF representation of KB ∧ ¬α
   new ← { }
   loop do
      for each Ci, Cj in clauses do
         resolvents ← PL-RESOLVE(Ci, Cj)
         if resolvents contains the empty clause then return true
         new ← new ∪ resolvents
      if new ⊆ clauses then return false
      clauses ← clauses ∪ new
```


Resolution [4]: Example

$KB = (B_{1,1} \Leftrightarrow (P_{1,2} \vee P_{2,1})) \wedge \neg B_{1,1}$ $\alpha = \neg P_{1,2}$



Chapter 7: Summary

Logical agents apply inference to a knowledge base
to derive new information and make decisions

Basic concepts of logic:
- syntax: formal structure of sentences
- semantics: truth of sentences wrt models
- entailment: necessary truth of one sentence given another
- inference: deriving sentences from other sentences
- soundess: derivations produce only entailed sentences
- completeness: derivations can produce all entailed sentences

Wumpus world requires the ability to represent partial and negated information, reason by cases, etc.

Forward, backward chaining are linear-time, complete for Horn clauses
Resolution is complete for propositional logic

Propositional logic lacks expressive power

Chapter 8: Overview

◇ Why FOL?

◇ Syntax and semantics of FOL

◇ Fun with sentences

◇ Wumpus world in FOL

Propositional Logic: Pros and Cons

- ☺ Propositional logic is **declarative**: pieces of syntax correspond to facts
- ☺ Propositional logic allows partial/disjunctive/negated information (unlike most data structures and databases)
- ☺ Propositional logic is **compositional**: meaning of $B_{1,1} \wedge P_{1,2}$ is derived from meaning of $B_{1,1}$ and of $P_{1,2}$
- ☺ Meaning in propositional logic is **context-independent** (unlike natural language, where meaning depends on context)
- ☹ Propositional logic has very limited expressive power (unlike natural language)
  E.g., cannot say "pits cause breezes in adjacent squares"
  except by writing one sentence for each square

First-Order Logic (FOL)

Whereas propositional logic assumes world contains **facts**,
first-order logic (like natural language) assumes the world contains

- Objects: people, houses, numbers, theories, Ronald McDonald, colors, baseball games, wars, centuries . . .
- Relations: red, round, bogus, prime, multistoried . . ., brother of, bigger than, inside, part of, has color, occurred after, owns, comes between, . . .
- Functions: father of, best friend, third inning of, one more than, end of . . .

Logics in General: Ontological and Epistemic Aspects

Language	Ontological Commitment	Epistemological Commitment
Propositional logic	facts	true/false/unknown
First-order logic	facts, objects, relations	true/false/unknown
Temporal logic	facts, objects, relations, times	true/false/unknown
Probability theory	facts	degree of belief
Fuzzy logic	facts + degree of truth	known interval value

Ontological commitment – what entities, relationships, and facts exist in world and can be reasoned about

Epistemic commitment – what agents can know about the world

Syntax of FOL: Basic Elements

Constants	$KingJohn,\ 2,\ UCB, \ldots$
Predicates	$Brother,\ >, \ldots$
Functions	$Sqrt,\ LeftLegOf, \ldots$
Variables	$x,\ y,\ a,\ b, \ldots$
Connectives	$\wedge\ \vee\ \neg\ \Rightarrow\ \Leftrightarrow$
Equality	$=$
Quantifiers	$\forall\ \exists$

Atomic Sentences (*aka* Atoms, *aka* Atomic WFFs)

Atomic sentence $=$ $predicate(term_1, \ldots, term_n)$
or $term_1 = term_2$

Term $=$ $function(term_1, \ldots, term_n)$
or $constant$ or $variable$

E.g., $Brother(KingJohn, RichardTheLionheart)$
$> (Length(LeftLegOf(Richard)), Length(LeftLegOf(KingJohn)))$

Atomic sentence – smallest unit of a logic
(*aka* "atom", "atomic well-formed formula (atomic WFF)")

Complex Sentences

Complex sentences are made from atomic sentences using connectives

$\neg S, \quad S_1 \wedge S_2, \quad S_1 \vee S_2, \quad S_1 \Rightarrow S_2, \quad S_1 \Leftrightarrow S_2$

E.g. $Sibling(KingJohn, Richard) \Rightarrow Sibling(Richard, KingJohn)$
$>(1,2) \vee \leq(1,2)$
$>(1,2) \wedge \neg{>}(1,2)$

Truth in First-Order Logic

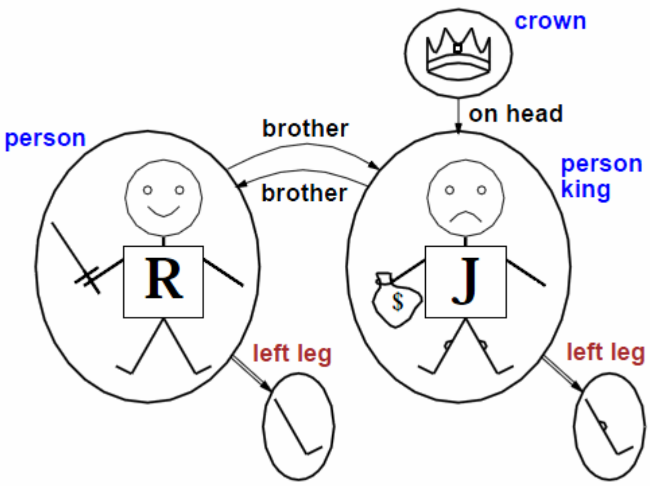
Sentences are true with respect to a model and an interpretation

Model contains ≥ 1 objects (domain elements) and relations among them

Interpretation specifies referents for
- constant symbols $\rightarrow$ objects
- predicate symbols $\rightarrow$ relations
- function symbols $\rightarrow$ functional relations

An atomic sentence $predicate(term_1, \ldots, term_n)$ is true
iff the objects referred to by $term_1, \ldots, term_n$
are in the relation referred to by $predicate$

Models for FOL: Example



Models for FOL: Example

Consider the interpretation in which
$Richard \rightarrow$ Richard the Lionheart
$John \rightarrow$ the evil King John
$Brother \rightarrow$ the brotherhood relation

Under this interpretation, $Brother(Richard, John)$ is true just in case Richard the Lionheart and the evil King John are in the brotherhood relation in the model

Models for FOL: Lots!

Entailment in propositional logic can be computed by enumerating models

We can enumerate the FOL models for a given KB vocabulary:

For each number of domain elements n from 1 to ∞
  For each k-ary predicate P_k in the vocabulary
    For each possible k-ary relation on n objects
      For each constant symbol C in the vocabulary
        For each choice of referent for C from n objects . . .

Computing entailment by enumerating FOL models is not easy!

Universal Quantification [1]: Definition

$\forall \langle variables \rangle \ \langle sentence \rangle$

Everyone at Berkeley is smart:
$\forall x \ \ At(x, Berkeley) \ \Rightarrow \ Smart(x)$

$\forall x \ \ P$ is true in a model m iff P is true with x being **each** possible object in the model

Roughly speaking, equivalent to the conjunction of instantiations of P

$$\begin{aligned} & (At(KingJohn, Berkeley) \ \Rightarrow \ Smart(KingJohn)) \\ \wedge \ & (At(Richard, Berkeley) \ \Rightarrow \ Smart(Richard)) \\ \wedge \ & (At(Berkeley, Berkeley) \ \Rightarrow \ Smart(Berkeley)) \\ \wedge \ & \ldots \end{aligned}$$

Universal Quantification [2]: Common Mistake to Avoid

Typically, $\Rightarrow$ is the main connective with $\forall$

Common mistake: using $\wedge$ as the main connective with $\forall$:

$$\forall x \ \ At(x, Berkeley) \wedge Smart(x)$$

means "Everyone is at Berkeley and everyone is smart"

Existential Quantification [1]: Definition

$\exists \langle variables \rangle \;\; \langle sentence \rangle$

Someone at Stanford is smart:
$\exists x \;\; At(x, Stanford) \land Smart(x)$

$\exists x \;\; P$ is true in a model m iff P is true with x being **some** possible object in the model

Roughly speaking, equivalent to the disjunction of instantiations of P

$$
\begin{array}{ll}
 & (At(KingJohn, Stanford) \land Smart(KingJohn)) \\
\lor & (At(Richard, Stanford) \land Smart(Richard)) \\
\lor & (At(Stanford, Stanford) \land Smart(Stanford)) \\
\lor & \ldots
\end{array}
$$

Existential Quantification [2]: Common Mistake to Avoid

Typically, $\land$ is the main connective with $\exists$

Common mistake: using $\Rightarrow$ as the main connective with $\exists$:

$$\exists x \;\; At(x, Stanford) \;\Rightarrow\; Smart(x)$$

is true if there is anyone who is not at Stanford!

Properties of Quantifiers

$\forall x\ \forall y$ is the same as $\forall y\ \forall x$ (why??)

$\exists x\ \exists y$ is the same as $\exists y\ \exists x$ (why??)

$\exists x\ \forall y$ is not the same as $\forall y\ \exists x$

$\exists x\ \forall y\ \ Loves(x, y)$
"There is a person who loves everyone in the world"

$\forall y\ \exists x\ \ Loves(x, y)$
"Everyone in the world is loved by at least one person"

Quantifier duality: each can be expressed using the other

$\forall x\ \ Likes(x, IceCream) \qquad \neg\exists x\ \ \neg Likes(x, IceCream)$

Fun With Sentences

Brothers are siblings

$\forall x, y\ \ Brother(x, y)\ \Rightarrow\ Sibling(x, y).$

"Sibling" is symmetric

$\forall x, y\ \ Sibling(x, y)\ \Leftrightarrow\ Sibling(y, x).$

One's mother is one's female parent

$\forall x, y\ \ Mother(x, y)\ \Leftrightarrow\ (Female(x) \land Parent(x, y)).$

A first cousin is a child of a parent's sibling

$\forall x, y\ \ FirstCousin(x, y)\ \Leftrightarrow\ \exists p, ps\ \ Parent(p, x) \land Sibling(ps, p) \land Parent(ps, y)$

Equality

$term_1 = term_2$ is true under a given interpretation
if and only if $term_1$ and $term_2$ refer to the same object

E.g., $1 = 2$ and $\forall x \;\; \times(Sqrt(x), Sqrt(x)) = x$ are satisfiable
$2 = 2$ is valid

E.g., definition of (full) $Sibling$ in terms of $Parent$:

$$\forall x, y \;\; Sibling(x,y) \;\Leftrightarrow\; [\neg(x = y) \wedge \exists m, f \;\; \neg(m = f) \wedge Parent(m,x) \wedge Parent(f,x) \wedge Parent(m,y) \wedge Parent(f,y)]$$

Interacting with FOL KBs

Suppose a wumpus-world agent is using an FOL KB
and perceives a smell and a breeze (but no glitter) at $t = 5$:

$Tell(KB, Percept([Smell, Breeze, None], 5))$
$Ask(KB, \exists a \;\; Action(a, 5))$

I.e., does KB entail any particular actions at $t = 5$?

Answer: $Yes, \;\; \{a/Shoot\}$ $\leftarrow$ substitution (binding list)

Given a sentence S and a substitution σ,
$S\sigma$ denotes the result of plugging σ into S; e.g.,
$S = Smarter(x, y)$
$\sigma = \{x/Hillary, y/Bill\}$
$S\sigma = Smarter(Hillary, Bill)$

$Ask(KB, S)$ returns some/all σ such that $KB \models S\sigma$

Knowledge Base for Wumpus World

"Perception"

$\forall\, b, g, t \;\; Percept([Smell, b, g], t) \;\Rightarrow\; Smelt(t)$

$\forall\, s, b, t \;\; Percept([s, b, Glitter], t) \;\Rightarrow\; AtGold(t)$

Reflex: $\forall\, t \;\; AtGold(t) \;\Rightarrow\; Action(Grab, t)$

Reflex with internal state: do we have the gold already?

$\forall\, t \;\; AtGold(t) \wedge \neg Holding(Gold, t) \;\Rightarrow\; Action(Grab, t)$

$Holding(Gold, t)$ cannot be observed

$\Rightarrow$ keeping track of change is essential

Deducing Hidden Properties

Properties of locations:

$\forall\, x, t \;\; At(Agent, x, t) \wedge Smelt(t) \;\Rightarrow\; Smelly(x)$

$\forall\, x, t \;\; At(Agent, x, t) \wedge Breeze(t) \;\Rightarrow\; Breezy(x)$

Squares are breezy near a pit:

Diagnostic rule—infer cause from effect

$\forall\, y \;\; Breezy(y) \;\Rightarrow\; \exists\, x \;\; Pit(x) \wedge Adjacent(x, y)$

Causal rule—infer effect from cause

$\forall\, x, y \;\; Pit(x) \wedge Adjacent(x, y) \;\Rightarrow\; Breezy(y)$

Neither of these is complete—e.g., the causal rule doesn't say whether squares far away from pits can be breezy

Terminology

- First-Order Logic (FOL) *aka* First-Order Predicate Calculus (FOPC)
  - Components
    - Semantics (meaning, denotation): objects, functions, relations
    - Syntax: constants, variables, terms, predicates
  - Properties of sentences (and sets of sentences, *aka* knowledge bases)
    - entailment
    - provability/derivability
    - validity: truth in all models (*aka* tautological truth)
    - satisfiability: truth in some models
  - Properties of proof rules
    - soundness: $KB \vdash_i \alpha \Rightarrow KB \models \alpha$ (can prove only true sentences)
    - completeness: $KB \models \alpha \Rightarrow KB \vdash_i \alpha$ (can prove all true sentences)
- Conjunctive Normal Form (CNF)
- Universal Quantification ("For All")
- Existential Quantification ("Exists")

Summary Points

- Last Class: Overview of Knowledge Representation (KR) and Logic
  - Representations covered in this course, by ontology and epistemology
  - Propositional calculus (*aka* propositional logic)
    - Syntax and semantics
    - Relationship to Boolean algebra
    - Properties
- Propositional Resolution
- Elements of Logics – Ontology, Epistemology
- Today: First-Order Logic (FOL) *aka* FOPC
  - Components: syntax, semantics
  - Sentences: entailment vs. provability/derivability, validity vs. satisfiability
  - Soundness and completeness
  - Properties of proof rules
    - soundness: $KB \vdash_i \alpha \Rightarrow KB \models \alpha$ (can prove only true sentences)
    - completeness: $KB \models \alpha \Rightarrow KB \vdash_i \alpha$ (can prove all true sentences)
- Next: First-Order Resolution