Logic and Reasoning

Foundations of Computing Science

Pallab Dasgupta
Professor,
Dept. of Computer Sc & Engg

Knowledge and Reasoning

- Representation, Reasoning and Logic
- Propositional Logic
- First-Order Logic
- Inference in first-order logic
  - Generalized Modus Ponens
  - Forward and backward chaining
  - Resolution
- Logical Reasoning Systems

The Wumpus World Environment

- Adjacent means left, right, top, or bottom
  - Stench: In squares containing and adjacent to wumpus
  - Breeze: In squares adjacent to a pit
- There can be one wumpus, one gold, and many pits. Agent starts from the bottom-left square of a grid.
- The agent dies if it enters a square containing a pit or the wumpus.
- The agent can shoot the wumpus along a straight line.
- The agent has only one arrow.

Logic

A formal system for describing states of affairs, consisting of:

- Syntax: describes how to make sentences, and
- Semantics: describes the relation between the sentences and the states of affairs

A proof theory – a set of rules for deducing the entailments of a set of sentences

Improper definition of logic, or an incorrect proof theory can result in absurd reasoning

Types of Logics

Language	What exists	Belief of agent
Propositional Logic	Facts	T / F / Unknown
First-Order Logic	Facts, Objects, Relations	T / F / Unknown
Temporal Logic	Facts, Objects, Relations, Times	T / F / Unknown
Probability Theory	Facts	Degree of belief [0..1]
Fuzzy Logic	Degree of truth	Degree of belief [0..1]

Propositional Logic

Given a set of atomic propositions $\mathcal{AP}$

- Sentence → Atom | ComplexSentence
- Atom → True | False | $\mathcal{AP}$
- ComplexSentence → (Sentence)
  | Sentence Connective Sentence
  | ¬ Sentence
- Connective → $\wedge \mid \vee \mid \Leftrightarrow \mid \Rightarrow$

Inference Rules

Modus Ponens or Implication Elimination:

$$\frac{\alpha \Rightarrow \beta, \quad \alpha}{\beta}$$

Unit Resolution:

$$\frac{\alpha \vee \beta, \quad \neg\beta}{\alpha}$$

Resolution:

$$\frac{\alpha \vee \beta, \quad \neg\beta \vee \gamma}{\alpha \vee \gamma}$$

or

$$\frac{\neg\alpha \Rightarrow \beta, \quad \beta \Rightarrow \gamma}{\neg\alpha \Rightarrow \gamma}$$

… and several other rules

Modeling in Propositional Logic

An Example (EX-1):

- **Proposition-1:** If the unicorn is mythical, then it is immortal, but if it is not mythical, then it is a mortal mammal.
- **Proposition-2:** If the unicorn is either immortal or a mammal, then it is horned.
- **Proposition-3:** The unicorn is magical if it is horned
- **Query:** Can we prove that the unicorn is mythical? Magical? Horned?

Propositions (EX-1):

- **Umyth:** Unicorn in mythical
- **Umort:** Unicorn is mortal
- **Umam:** Unicorn is mammal
- **Umag:** Unicorn is magical
- **Uhorn:** Unicorn is horned

Automated Reasoning

In general, the inference problem is NP-complete (Cook's Theorem)

If we restrict ourselves to Horn sentences, then repeated use of Modus Ponens gives us a poly-time procedure. Horn sentences are of the form:

$$\boxed{P1 \land P2 \land \ldots \land Pn \rightarrow Q}$$

First-Order Logic

Constant → A | 5 | Kolkata | …

Variable → a | x | s | …

Predicate → Before | HasColor | Raining | …

Function → Mother | Cosine | Headoflist | …

Sentence → AtomicSentence
| Sentence Connective Sentence
| Quantifier Variable, … Sentence
| ¬ Sentence | (Sentence)

AtomicSentence → Predicate(Term, …) | Term = Term

Term → Function(Term, …) | Constant | Variable

Connective → Λ | V | => | <=>

Quantifier → ∀| ∃

Examples

Not all students take both History & Biology

Only one student failed History

Only one student failed both History & Biology

The best score in History is better than the best score in Biology

No person likes a professor unless the professor is smart

Politicians can fool some of the people all the time, and they can fool all the people some of the time, but they cant fool all the people all the time

Russel's Paradox:

- There is a single barber in town.
- Those and only those who do not shave themselves are shaved by the barber.
- Who shaves the barber?

Inference Rules

Universal elimination:

- $\forall x$ Likes(x, IceCream) with the substitution {x / Einstein} gives us Likes(Einstein, IceCream)
- The substitution has to be done by a ground term

Existential elimination:

- From $\exists x$ Likes(x, IceCream) we may infer Likes(Man, IceCream) as long as Man does not appear elsewhere in the Knowledge base

Existential introduction:

- From Likes(Monalisa, IceCream) we can infer $\exists x$ Likes(x, IceCream)

Reasoning in First-Order Logic

Example:

- The law says that it is a crime for a Gaul to sell potion formulas to hostile nations.
- The country Rome, an enemy of Gaul, has acquired some potion formulas, and all of its formulas were sold to it by Druid Traitorix.
- Traitorix is a Gaul.
- Is Traitorix a criminal?

Generalized Modus Ponens

For atomic sentences p_i, p_i', and q, where there is a substitution θ such that

SUBST(θ, p_i') = SUBST(θ, p_i), for all i:

$$\frac{p_1', p_2', \ldots, p_n', (p_1 \wedge p_2 \wedge \ldots \wedge p_n \Rightarrow q)}{SUBST(\theta, q)}$$

Unification

$UNIFY(p,q) = \theta$ where $SUBST(\theta,p) = SUBST(\theta,q)$

Examples-1:

- UNIFY(Knows(Erdos, x),Knows(Erdos, Godel)) = {x / Godel}
- UNIFY(Knows(Erdos, x), Knows(y,Godel)) = {x/Godel, y/Erdos}

Examples-2:

- UNIFY(Knows(Erdos, x), Knows(y, Father(y))) = { y/Erdos, x/Father(Erdos) }
- UNIFY(Knows(Erdos, x), Knows(x, Godel)) = F

We require the most general unifier

Reasoning with Horn Logic

We can convert Horn sentences to a canonical form and then use generalized Modus Ponens with unification.

- We skolemize existential formulas and remove the universal ones
- This gives us a conjunction of clauses, that are inserted in the KB
- Modus Ponens help us in inferring new clauses

Forward and backward chaining

Completeness Issues

Reasoning with Modus Ponens is incomplete

Consider the example –

$$\forall x\ P(x) \Rightarrow Q(x) \qquad \forall x\ \neg P(x) \Rightarrow R(x)$$

$$\forall x\ Q(x) \Rightarrow S(x) \qquad \forall x\ R(x) \Rightarrow S(x)$$

We should be able to conclude S(A)

The problem is that $\forall x\ \neg P(x) \Rightarrow R(x)$ cannot be converted to Horn form, and thus cannot be used by Modus Ponens

Godel’s Completeness Theorem

For first-order logic, any sentence that is entailed by another set of sentences can be proved from that set

- Godel did not suggest a proof procedure
- In 1965 Robinson published his resolution algorithm

Entailment in first-order logic is semi-decidable, that is, we can show that sentences follow from premises if they do, but we cannot always show if they do not.

The Validity Problem of First-Order Logic

[Church] The validity problem of the first-order predicate calculus is partially solvable.

Consider the following formula:

$$[\bigwedge_{i=1}^{n} p(f_i(a), g_i(a))$$
$$\wedge \forall x \forall y [p(x, y) \Rightarrow \bigwedge_{i=1}^{n} p(f_i(x), g_i(y))]]$$
$$\Rightarrow \exists z \; p(z, z)$$

Resolution

Generalized Resolution Rule:

For atoms p_i, q_i, r_i, s_i, where Unify(p_j, q_k) = θ, we have:

$$\frac{\begin{array}{c} p_1 \wedge \ldots p_j \ldots \wedge p_{n1} \Rightarrow r_1 \vee \ldots r_{n2} \\ s_1 \wedge \ldots \wedge s_{n3} \Rightarrow q_1 \vee \ldots q_k \ldots \vee q_{n4} \end{array}}{\begin{array}{l} \mathrm{SUBST}(\theta, \\ \quad p_1 \wedge \ldots p_{j-1} \wedge p_{j+1} \ldots \wedge p_{n1} \wedge s_1 \wedge \ldots s_{n3} \\ \quad\quad \Rightarrow r_1 \vee \ldots r_{n2} \vee \ldots q_{k-1} \vee q_{k+1} \vee \ldots \vee q_{n4}) \end{array}}$$

An Example

$$P(w) \Rightarrow Q(w)$$

$$Q(y) \Rightarrow S(y)$$

$$\{y / w\}$$

$$P(w) \Rightarrow S(w)$$

$$True \Rightarrow P(x) \vee R(x)$$

$$\{w / x\}$$

$$True \Rightarrow S(x) \vee R(x)$$

$$R(z) \Rightarrow S(z)$$

$$\{x/A, z/A\}$$

$$True \Rightarrow S(A)$$

The Basic Steps

Convert the set of rules and facts into clause form (conjunction of clauses)

Insert the negation of the goal as another clause

Use resolution to deduce a refutation

If a refutation is obtained, then the goal can be deduced from the set of facts and rules.

Conversion to Normal Form

A formula is said to be in clause form if it is of the form:

$$\forall x_1\ \forall x_2 \ldots \forall x_n\ [C_1 \wedge C_2 \wedge \ldots \wedge C_k]$$

All first-order logic formulas can be converted to clause form

We shall demonstrate the conversion on the formula:

$$\boxed{\begin{aligned} &\forall x\ \{p(x) \Rightarrow \exists z\ \{\ \neg\forall y\ [q(x,y) \Rightarrow p(f(x_1))] \\ &\qquad\qquad\qquad\qquad \wedge \forall y\ [q(x,y) \Rightarrow p(x)]\ \}\} \end{aligned}}$$

Conversion to Normal Form

Step1: ***Take the existential closure and eliminate redundant quantifiers.*** **This introduces $\exists x_1$ and eliminates $\exists z$, so:**

$$\forall x \{p(x) \Rightarrow \exists z \{ \neg\forall y [q(x,y) \Rightarrow p(f(x_1))] \wedge \forall y [q(x,y) \Rightarrow p(x)] \}\}$$

$$\exists x_1 \forall x \{p(x) \Rightarrow \{ \neg\forall y [q(x,y) \Rightarrow p(f(x_1))] \wedge \forall y [q(x,y) \Rightarrow p(x)] \}\}$$

Conversion to Normal Form

Step 2: ***Rename any variable that is quantified more than once.*** **y has been quantified twice, so:**

$$\exists x_1 \, \forall x \, \{p(x) \Rightarrow \{ \neg \forall y \, [q(x,y) \Rightarrow p(f(x_1))] \wedge \forall y \, [q(x,y) \Rightarrow p(x)] \}\}$$

$$\exists x_1 \, \forall x \, \{p(x) \Rightarrow \{ \neg \forall y \, [q(x,y) \Rightarrow p(f(x_1))] \wedge \forall z \, [q(x,z) \Rightarrow p(x)] \}\}$$

Conversion to Normal Form

Step 3: ***Eliminate implication.***

$$\exists x_1\ \forall x\ \{p(x) \Rightarrow \{\neg\forall y\ [q(x,y) \Rightarrow p(f(x_1))]$$
$$\wedge\ \forall z\ [q(x,z) \Rightarrow p(x)]\ \}\}$$

$$\exists x_1\ \forall x\ \{\neg p(x) \vee \{\neg\forall y\ [\neg q(x,y) \vee p(f(x_1))]$$
$$\wedge\ \forall z\ [\neg q(x,z) \vee p(x)]\ \}\}$$

Conversion to Normal Form

Step 4: ***Move $\neg$ all the way inwards.***

$$\exists x_1\ \forall x\ \{\neg p(x) \vee \{\ \neg \forall y\ [\neg q(x,y) \vee p(f(x_1))] \wedge \forall z\ [\neg q(x,z) \vee p(x)]\ \}\}$$

$$\exists x_1\ \forall x\ \{\neg p(x) \vee \{\exists y\ [q(x,y) \wedge \neg p(f(x_1))] \wedge \forall z\ [\neg q(x,z) \vee p(x)]\ \}\}$$

Conversion to Normal Form

Step 5: ***Push the quantifiers to the right.***

$$\exists x_1 \, \forall x \, \{\neg p(x) \lor \{\exists y \, [q(x,y) \land \neg p(f(x_1))] \land \forall z \, [\neg q(x,z) \lor p(x)] \}\}$$

$$\exists x_1 \, \forall x \, \{\neg p(x) \lor \{[\exists y \, q(x,y) \land \neg p(f(x_1))] \land [\forall z \, \neg q(x,z) \lor p(x)] \}\}$$

Conversion to Normal Form

Step 6: ***Eliminate existential quantifiers (Skolemization).***

- Pick out the leftmost $\exists y\, B(y)$ and replace it by $B(f(x_{i1}, x_{i2}, \ldots, x_{in}))$, where:
  a) $x_{i1}, x_{i2}, \ldots, x_{in}$ are all the distinct free variables of $\exists y\, B(y)$ that are universally quantified to the left of $\exists y\, B(y)$, and
  b) F is any n-ary function constant which does not occur already

Skolemization:

$$\exists x_1\ \forall x\ \{\neg p(x) \vee \{[\exists y\ q(x,y) \wedge \neg p(f(x_1))] \wedge [\forall z\ \neg q(x,z) \vee p(x)]\}\}$$

$$\forall x\ \{\neg p(x) \vee \{[q(x,g(x)) \wedge \neg p(f(a))] \wedge [\forall z\ \neg q(x,z) \vee p(x)]\}\}$$

Conversion to Normal Form

Step 7: ***Move all universal quantifiers to the left***

$$\forall x\ \{\neg p(x) \vee \{[q(x,g(x)) \wedge \neg p(f(a))] \\ \wedge [\forall z\ \neg q(x,z) \vee p(x)]\ \}\}$$

$$\forall x\ \forall z\ \{\neg p(x) \vee \{[q(x,g(x)) \wedge \neg p(f(a))] \\ \wedge [\neg q(x,z) \vee p(x)]\ \}\}$$

Conversion to Normal Form

Step 8: ***Distribute $\wedge$ over $\vee$.***

$$
\begin{aligned}
\forall x\ \forall z\ \{ & [\neg p(x) \vee q(x,g(x))] \\
& \wedge [\neg p(x) \vee \neg p(f(a))] \\
& \wedge [\neg p(x) \vee \neg q(x,z) \vee p(x)]\ \}
\end{aligned}
$$

Step 9: **(Optional)** ***Simplify***

$$\forall x\ \{[\neg p(x) \vee q(x,g(x))] \ \wedge\ \neg p(f(a))\ \}$$

Resolution

If Unify(z_j, $\neg q_k$) = θ, then:

$$\frac{z_1 \vee \ldots \vee z_m, \qquad q_1 \vee \ldots \vee q_n}{\mathrm{SUBST}(\theta, z_1 \vee \ldots \vee z_{i-1} \vee z_{i+1} \vee \ldots \vee z_m \vee q_1 \vee \ldots \vee q_{k-1} \vee q_{k+1} \vee \ldots \vee q_n)}$$

Example

Harry, Ron and Draco are students of the Hogwarts school for wizards

Every student is either wicked or is a good Quidditch player, or both

No Quidditch player likes rain and all wicked students like potions

Draco dislikes whatever Harry likes and likes whatever Harry dislikes

Draco likes rain and potions

Is there a student who is good in Quidditch but not in potions?

Example

Student(Harry)
Student(Ron)
Student(Draco)

$\forall x$ [Student(x) $\Rightarrow$ Wicked(x) $\lor$ Player(x)]
$\forall x$ [Player(x) $\Rightarrow \neg$ Likes(x, Rain)]
$\forall x$ [Student(x) $\land$ Wicked(x) $\Rightarrow$ Likes(x, Potions)]
$\forall x$ [Likes(Harry, x) $\Rightarrow \neg$ Likes(Draco, x)]
$\forall x$ [$\neg$ Likes(Harry, x) $\Rightarrow$ Likes(Draco, x)]

Likes(Draco, Rain)
Likes(Draco, Potions)

Goal: $\exists$ Student(x) $\land$ Player(x) $\land \neg$ Likes(x, Potions)

C1: Student(Harry)
C2: Student(Ron)
C3: Student(Draco)

C4: $\neg$ Student(x) $\lor$ Wicked(x) $\lor$ Player(x)
C5: $\neg$ Player(y) $\lor \neg$ Likes(y, Rain)
C6: $\neg$ Student(x) $\lor \neg$ Wicked(x) $\lor$ Likes(x, Potions)
C7: $\neg$ Likes(Harry, x) $\lor \neg$ Likes(Draco, x)
C8: $\neg$ Likes(Harry, x) $\lor$ Likes(Draco, x)

C9: Likes(Draco, Rain)
C10: Likes(Draco, Potions)

Negation of Goal:
C11: $\neg$ Student(x) $\lor \neg$ Player(x) $\lor$ Likes(x, Potions)

Resolution Strategies

Unit Resolution

- Every resolution step must involve a unit clause
- Leads to a good speedup
- Incomplete in general
- Complete for Horn knowledge bases

Input Resolution

- Every resolution step must involve a input sentence (from the query or the KB)
- In Horn knowledge bases, Modus Ponens is a kind of input resolution strategy
- Incomplete in general
- Complete for Horn knowledge bases

Resolution Strategies

Linear Resolution

- Slight generalization of input resolution
- Allows P and Q to be resolved together either if P is in the original KB, or if P is an ancestor of Q in the proof tree
- Linear resolution is complete

Incompleteness of Input Resolution

Use Linear Resolution to obtain a refutation:

$\forall x \, [\, P(x) \vee Q(x) \,]$
$\forall x \, [\, \neg P(x) \vee \neg Q(x) \,]$
$P(a) \vee \neg Q(a)$
$\neg P(a) \vee Q(a)$