Problem Session Week 9

Trevor Maxfield
maxfit@stanford.edu

Minae Kwon
minae@cs.stanford.edu

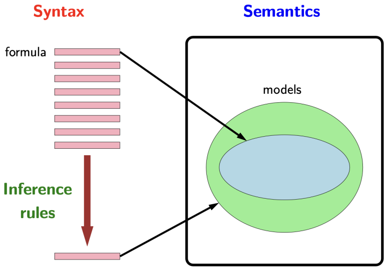
June Second, 2023

CS 221 - Spring 2023, Stanford University

Reviewing Lecture Material

Logic

- Thinking in terms of logical formulas and inference rules, as opposed to **state** or **variable** based models.
- "Logic language" to represent and reason with knowledge.
  - Syntax: defines valid formulas.
  - Semantics: specify models (satisfying assignments) for each formula.
  - Inference rules: what does a formula imply?



Reviewing Lecture Material

Propositional Logic

Syntax and Semantics of Propositional Logic

Syntax

- Propositional symbols: A, B, C, Cat, ...
  - Can be anything
- Logical connectives: $\neg, \wedge, \vee, \rightarrow, \leftrightarrow$
  - Not, And, Or, Implies, Equals
- Build up formulas recursively, can operate on formulas with logical connectives

Semantics

Definition
A model in propositional logic is an **assignment** of truth values to propositional symbols.

Interpretation function: if model w satisfies formula f, then

$$\mathcal{I}(f, w) \in \{0, 1\}$$

Models

- Let $\mathcal{M}(f)$ be the set of models w for which $\mathcal{I}(f, w) = 1$.
  - Set of all possible valid assignments. A formula compactly represents a set of models.
- A knowledge base KB is a set of formulas representing their intersection:

$$\mathcal{M}(KB) = \bigcap_{f \in KB} \mathcal{M}(f)$$

  - KB specifics constraints on the world, $\mathcal{M}(KB)$ is the set of all worlds satisfying those constraints.
  - Remember $\mathcal{M}(f)$ is just a set of models, this is intersection of those sets over a number of f.
- Adding more formulas to the knowledge base ...

Models

- Let $\mathcal{M}(f)$ be the set of models w for which $\mathcal{I}(f, w) = 1$.
  - Set of all possible valid assignments. A formula compactly represents a set of models.
- A knowledge base KB is a set of formulas representing their intersection:

$$\mathcal{M}(KB) = \bigcap_{f \in KB} \mathcal{M}(f)$$

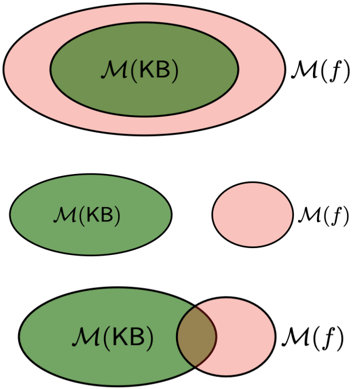
  - KB specifics constraints on the world, $\mathcal{M}(KB)$ is the set of all worlds satisfying those constraints.
  - Remember $\mathcal{M}(f)$ is just a set of models, this is intersection of those sets over a number of f.
- Adding more formulas to the knowledge base ... shrinks the set of models

Adding Formulas

When we add f to KB:

1. Entailment: no information was added.
   - $\text{KB} \models f$ iff $\mathcal{M}(\text{KB}) \subseteq \mathcal{M}(f)$
2. Contradiction: f contradicts what we know
   - KB contradicts f iff $\mathcal{M}(\text{KB}) \cap \mathcal{M}(f) = \emptyset$
3. Contingency: f adds non-trivial information to *KB*
   - $\emptyset \subsetneq \mathcal{M}(\text{KB}) \cap \mathcal{M}(f) \subsetneq \mathcal{M}(\text{KB})$

KB contradicts f iff KB entails $\neg f$.



Inference Rules

Inference rules allow us to reason with formulas without ever instantiating models.

Modus Ponens: for any propositional symbols p and q:

$$\frac{p, \quad p \rightarrow q}{q} \qquad \text{which is } \frac{\text{(premises)}}{\text{(conclusion)}}$$

Inference rule: (syntax, not semantics!) if $f_1, \ldots, f_k, g$ are formulas then:

$$\frac{f_1, \ \ldots \ , f_k}{g}$$

KB derives/proves f (KB $\vdash f$) iff f eventually is added to KB through inference rules.

Conjunctive Normal Form

A CNF formula is a conjunction (and) of clauses (or's):

$$(A \vee B \vee \neg C) \wedge (\neg B \vee D)$$

Can always convert:

- $f \leftrightarrow g$ is $(f \rightarrow g) \wedge (g \rightarrow f)$
- $f \rightarrow g$ is $\neg f \vee g$
- $\neg(f \wedge g)$ is $\neg f \vee \neg g$
- $\neg(f \vee g)$ is $\neg f \wedge \neg g$
- Double negatives cancel.
- Can distribute $\vee$ over $\wedge$: $f \vee (g \wedge h)$ is $(f \vee g) \wedge (f \vee h)$

Resolution

Remember that entailment $\text{KB} \models f$ is the opposite of contradiction $\text{KB} \cup \{\neg f\}$ is unsatisfiable.

Resolution-based inference:

- Add $\neg f$ into KB
- Convert all formulas into CNF
- Repeatedly apply resolution rule.
- Return entailment iff derive false.

Resolution rule:

$$\frac{f_1 \vee \cdots \vee f_n \vee p, \quad \neg p \vee g_1 \vee \cdots \vee g_m}{f_1 \vee \cdots \vee f_n \vee g_1 \vee \cdots \vee g_m}$$

Reviewing Lecture Material

First Order Logic

First Order Logic

Syntax

Uses terms to refer to objects:

- Constant symbols (e.g. arithmetic)
- Variable (e.g. x)
- Functions of terms (e.g. $\mathrm{Sum}(3, x)$)

Formulas refer to truth values:

- Atomic formulas (atoms), predicate applied to terms.
  - $\mathrm{Knows}(x, \mathit{arithmetic})$
- Connectives applied to formulas:
  - $\mathrm{Student}(x) \rightarrow \mathrm{Knows}(x, \mathrm{arithmetic})$
- Quantifiers applied to formulas:
  - $\forall x \mathrm{Student}(x) \rightarrow \mathrm{Knows}\ (x, \mathrm{arithmetic})$

Quantifiers

- Universal quantification ($\forall$):
  - Like conjunction (and): $\forall x P(x)$ is $P(A) \wedge P(B) \wedge \cdots$
- Existential quantification ($\exists$):
  - Like disjunction (or): $\exists x P(x)$ is like $P(A) \vee P(B) \vee \cdots$

  Properties:
  - $\neg \forall x P(x)$ is equivalently $\exists x \neq P(x)$
  - $\forall x \exists y \text{Knows}(x, y)$ is different from $\exists y \forall x \text{Knows}(x, y)$.

Models in first order logic map constant symbols to objects and predicate symbols to (satisfying) tuples of objects.

Summary

Things we Skipped Reviewing

- Soundness and completeness
- Propositional Horn clauses (definite + goal clause)
- First order modens ponens
- First order resolution

Problems

Reviewing Lecture Material

Propositional Logic

First Order Logic

Summary

Problems

Definite Clauses (Propositional)

Definition
A definite clause has the following form:

$$(p_1 \wedge \cdots \wedge p_k) \rightarrow q$$

p_i and q are propositional symbols. Formula, not an inference rule.

A Horn clause is either:

- A definite clause $(p_1 \wedge \cdots \wedge p_k) \rightarrow q$
- A goal clause $(p_1 \wedge \cdots \wedge p_k) \rightarrow \text{false}$, equivalently $\neg(p_1 \wedge \cdots \wedge p_k)$.