Knowledge Representation III
First-Order Logic

CSE 473

Logic-Based KR

- Propositional logic
  - Syntax (CNF, Horn clauses, ...)
  - Semantics (Truth Tables)
  - Inference (FC, Resolution, DPLL, WalkSAT)
  - Restricted Subsets
- First-order logic
  - Syntax (quantifiers, skolem functions, ...)
  - Semantics (Interpretations)
  - Inference (FC, Resolution, Compilation)
  - Restricted Subsets (e.g. Frame Systems)
- Representing events, action & change

Propositional. Logic vs. First Order

<table>
<thead>
<tr>
<th>Ontology</th>
<th>Facts (P, Q)</th>
<th>Objects, Properties, Relations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Syntax</td>
<td>Atomic sentence Connectives</td>
<td>Variables &amp; quantification Sentences have structure: terms father-of(mother-of(X))</td>
</tr>
<tr>
<td>Semantics</td>
<td>Truth Tables</td>
<td>Interpretations (Much more complicated)</td>
</tr>
<tr>
<td>Inference</td>
<td>DPLL, GSAT</td>
<td>Unification</td>
</tr>
<tr>
<td>Algorithm</td>
<td>Fast in practice</td>
<td>Forward, Backward chaining</td>
</tr>
<tr>
<td>Complexity</td>
<td>NP-Complete</td>
<td>Semi-decidable</td>
</tr>
</tbody>
</table>

FOL Definitions

- **Constants**: a, b, dog33.
  - Name a specific object.
- **Variables**: X, Y.
  - Refer to an object without naming it.
- **Functions**: dad-of
  - Mapping from objects to objects.
- **Terms**: dad-of(dog33)
  - Refer to objects
- **Atomic Sentences**: in(dad-of(dog33), food6)
  - Can be true or false
  - Correspond to propositional symbols P, Q

More Definitions

- **Logical connectives**: and, or, not, =>
- **Quantifiers**:
  - ∀ Forall
  - ∃ There exists
- **Examples**
  - Dumbo is grey
  - Elephants are grey
  - There is a grey elephant
Quantifier / Connective Interaction

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1. $\forall x \ E(x) \land G(x)$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2. $\forall x \ E(x) \Rightarrow G(x)$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3. $\exists x \ E(x) \land G(x)$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4. $\exists x \ E(x) \Rightarrow G(x)$</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

E(x) == “x is an elephant”
G(x) == “x has the color grey”

Nested Quantifiers: Order matters!

$\forall x \exists y \ P(x,y) \neq \exists y \forall x \ P(x,y)$

Examples

- Every dog has a tail
- Every dog shares a tail?

Someone is loved by everyone

$\exists x \forall y \ loves(y, x)$

Semantics

- Syntax: a description of the legal arrangements of symbols (Def “sentences”)
- Semantics: what the arrangement of symbols means in the world

Propositional Logic: SEMANTICS

- “Interpretation” (or “possible world”) Specifically, TRUTH TABLES
- Assignment to each variable either T or F
- Assignment of T or F to each connective via defns

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>P</td>
<td>T</td>
<td>F</td>
</tr>
<tr>
<td>Q</td>
<td>T</td>
<td>F</td>
</tr>
</tbody>
</table>

$P \land Q$SEMANTICS

Interpretations=Mappings
syntactic tokens → model elements

Depiction of one possible interpretation, assuming

- Constants: Richard, John
- Functions: Leg(p, l)
- Relations: On(x, y), King(p)
Interpretations = Mappings
syntactic tokens → model elements

Another interpretation, same assumptions

Constants: Richard John
Functions: Leg(p,l) On(x,y)
Relations: King(p)

Satisfiability, Validity, & Entailment

• S is valid if it is true in all interpretations
• S is satisfiable if it is true in some interp
• S is unsatisfiable if it is false all interps

|= S1 entails S2 if
forall interps where S1 is true,
S2 is also true

Skolemization

• Existential quantifiers aren't necessary!
  Existential variables can be replaced by
  • Skolem functions (or constants)
  • Args to function are all surrounding ∀ vars

• ∀d ∃t has(d, t) → ∀d has(d, f(d))
• ∃x ∀y loves(y, x) → ∀y loves(y, f())

FOL Reasoning

• FO Forward & Backward Chaining
• FO Resolution
• Many other types of theorem proving
• Restricted representations
  Description logics
• Compilation to SAT

Forward Chaining

• Given
  ∀?x lifeform(?x) => mortal(?x)
  ∀?x mammal(?x) => lifeform(?x)
  ∀?x dog(?x) => mammal(?x)
dog(fido)

• Prove
  mortal(fido)

∀?x dog(?x) => mammal(?x)
   ________________ ?
dog(fido)
mammal(fido)

Unification

• Emphasize variables with ?
• Useful for FO inference (modus ponens, ...)
  Also for compilation of FOPC -> propositional

• Unify(x, y) returns "mgu"
  Unify(city(?a), city(kent)) returns ?a/kent

• Substitute(expr, mapping) returns new expr
  Substitute(connected(?a, ?b), (?a/kent)) returns connected(kent, ?b)
Unification Examples

• Unify(road(?a, kent), road(seattle, ?b))
• Unify(road(?a, ?a), road(seattle, kent))
• Unify(f(g(?x, dog), ?y)), f(g(cat, ?y), dog)
• Unify(f(g(?x)), f(?x))

Resolution

[Robinson 1965]

\{ (p \lor \alpha), (\neg p \lor \beta \lor \gamma) \} \vdash (\alpha \lor \beta \lor \gamma)

Recall Propositional Case:
• Literal in one clause
• Its negation in the other
• Result is disjunction of other literals

First-Order Resolution

[Robinson 1965]

\{ (p(?x) \lor a(a), \neg p(q) \lor b(?x) \lor c(?y)) \}
\vdash (a(a) \lor b(q) \lor c(?y))

• Literal in one clause
• The negation of something which unifies in the other
• Result is disjunction of other literals / mgu

Example

• Given
  \forall x \text{ man}(?x) \rightarrow \text{mortal}(?x)
  \forall x \text{ woman}(?x) \rightarrow \text{mortal}(?x)
  \forall x \text{ person}(?x) \rightarrow \text{man}(?x) \lor \text{woman}(?x)
  \text{person(kelly)}
• Prove
  \text{mortal(kelly)}

[\neg \text{m(?x),d(?x)}] [\neg \text{w(?y),d(?y)}] [\neg \text{p(?z),m(?z),w(?z)}] [p (k)] [\neg \text{d(k)}]

Example Continued

[\neg \text{m(?x),d(?x)}] [\neg \text{w(?y),d(?y)}] [\neg \text{p(?z),m(?z),w(?z)}] [p (k)] [\neg \text{d(k)}]

[m(k),w(k)]
[w(k), d(k)]
[d(k)]

[w(k)]
[d(k)]

KR with Description Logics

**Tbox**
- Term definitions
- FO Language + inference organized into a taxonomy, e.g:
  - father\(x\) = person\(x\) ∧ male\(x\) ∧ \(\exists y\) childof\(y, x\)
  - parent\(x\) = person\(x\) ∧ \(\exists y\) childof\(y, x\)
- Complexity of classifying new terms
  - subsumption

Subsumption hierarchy

**Abox**
- Assertions
- Abox - separate language + inference for "propositional" assertions using Tbox terms
  - e.g. person(kelley)

**Debate**
- Restricted language thesis
  - Disjunction, negation, particularization, order...
  - Natural kinds
- Restricted classification thesis
  - Concepts using contingent information:
    - Treatable disease, democratic country, illegal act
- Counterargument
  - Constructs: Omit vs limit
    - Completeness
    - Efficiency

**Compilation to Prop. Logic I**
- Typed Logic
  - ∀\(city\) a, b connected\(a, b\)
- Universe
  - Cities: seattle, tacoma, enumclaw
- Equivalent propositional formula:

**Compilation to Prop. Logic II**
- Universe
  - Cities: seattle, tacoma, enumclaw
  - Firms: IBM, Microsoft, Boeing
- First-Order formula
  - ∀\(city\) c \(\exists f_{firm}\) f hasHQ\(c, f\)
- Equivalent propositional formula
Hey!

- You said FO Inference is semi-decidable
- But you compiled it to SAT
  Which is NP Complete
- So now we can always do the inference?!
  Tho it might take exponential time...
- Something seems wrong here....????

Restricted Forms of FO Logic

- Known, Finite Universes
  Compile to SAT
- Frame Systems
  Ban certain types of expressions
- Horn Clauses
  Aka Prolog
- Function-Free Horn Clauses
  Aka Datalog