Intelligent Systems (AI-2)

Computer Science cpsc422, Lecture 22

Mar, 10, 2021

Slide credit: some from Prof. Carla P. Gomes (Cornell) some slides adapted from Stuart Russell (Berkeley), some from Prof. Jim Martin (Univ. of Colorado)
Relationships between different Logics (better with colors)

First Order Logic

\( \forall X \exists Y p(X, Y) \iff \neg q(Y) \)

- \( p(a_1, a_2) \)
- \( \neg q(a_3) \)

Propositional Logic

\( \neg (p \lor q) \iff (r \land s \land \neg t) \)

- \( p, r \)

Datalog

- \( p(X) \leftarrow q(X) \land r(X, Y) \)
- \( r(X, Y) \leftarrow s(Y) \)
- \( s(a_1), q(a_2) \)

PDCL

- \( p \leftarrow s \land t \\
- \( r \leftarrow s \land q \land p \\
- \( r \leftarrow p \)
Lecture Overview

• SAT: example
• First Order Logics
  • Language and Semantics
  • Inference
Satisfiability problems (SAT)

Consider a CNF sentence, e.g.,

$$(\neg D \lor \neg B \lor C) \land (B \lor \neg A \lor \neg C) \land (\neg C \lor \neg B \lor E) \land (E \lor \neg D \lor B) \land (B \lor E \lor \neg C)$$

Is there an interpretation in which this sentence is true (i.e., that is a model of this sentence)?

Many combinatorial problems can be reduced to checking the satisfiability of propositional sentences …..and returning a model
Encoding the Latin Square Problem in Propositional Logic

In combinatorics and in experimental design, a Latin square is

- an \( n \times n \) array
- filled with \( n \) different symbols,
- each occurring exactly once in each row and exactly once in each column.

Here is an example:

\[
\begin{array}{ccc}
A & B & C \\
C & A & B \\
B & C & A \\
\end{array}
\]

Here is another one:
Encoding Latin Square in Propositional Logic:

Propositions

Variables must be binary! (They must be propositions)

Each variables represents a color assigned to a cell $ij$.

Assume colors are encoded as an integer $k$

$$x_{ijk} \in \{0, 1\}$$

Assuming colors are encoded as follows

(black, 1) (red, 2) (blue, 3) (green, 4) (purple, 5)

True or false, ie. 1 or 0 with respect to the interpretation represented by the picture?

How many vars/propositions overall?
Encoding Latin Square in Propositional Logic: Clauses

- Some color must be assigned to each cell (clause of length 5);

\[ \forall_{ij} (x_{ij1} \lor x_{ij2} \ldots x_{ij5}) \]

\[ \forall_{ik} (x_{i1k} \lor x_{i2k} \ldots x_{i5k}) \]

How many clauses?
Encoding Latin Square in Propositional Logic: Clauses

- Some color must be assigned to each cell (clause of length n);

\[ \forall_{ij} (x_{ij1} \lor x_{ij2} \cdots x_{ijn}) \quad \forall_{ik} (x_{ilk} \lor x_{i2k} \cdots x_{ink}) \quad \forall_{ik} (\neg x_{ilk} \lor \neg x_{i2k} \cdots \neg x_{ink}) \cdots (\neg x_{ilk} \lor \neg x_{i3k} \cdots (\neg x_{ilk} \lor \neg x_{ink}) \cdots (\neg x_{ilk} \lor \neg x_{i(n-1)k} \lor \neg x_{ink}) \]

- No color is repeated in the same row (sets of negative binary clauses);

How many clauses?
Encoding Latin Square in Propositional Logic: Clauses

- Some color must be assigned to each cell (clause of length n);

\[ \forall ij (x_{ij1} \lor x_{ij2} \ldots x_{ijn}) \]

- No color repeated in the same row (sets of negative binary clauses);

\[ \forall ik (\neg x_{i1k} \lor \neg x_{i2k}) \land (\neg x_{i1k} \lor \neg x_{i3k}) \land \ldots \land (\neg x_{i1k} \lor \neg x_{ink}) \land (\neg x_{i(n-1)k} \lor \neg x_{ink}) \]

How many clauses?

\[ \frac{n^2 \times n \times (n-1)}{2} \]

\[ O(n^4) \]
Encoding Latin Square Problems in Propositional Logic: FULL MODEL

- **Variables:**
  \[ x_{ijk} \text{ cell } i, j \text{ has color } k; \quad i,j,k=1,2,...,n. \]
  \[ x_{ijk} \in \{0,1\} \]
  Each variable represents a color assigned to a cell.

- **Clauses:** \( O(n^4) \)
  - Some color must be assigned to each cell (clause of length n);
    \[ \forall ij (x_{ij1} \lor x_{ij2} \ldots x_{ijn}) \]
  - No color repeated in the same row (sets of negative binary clauses);
    \[ \forall ik (\neg x_{ilk} \lor \neg x_{i2k}) \land (\neg x_{ilk} \lor \neg x_{i3k}) \ldots (\neg x_{ilk} \lor \neg x_{ink}) \ldots (\neg x_{ilk} \lor \neg x_{i(n-1)k} \lor \neg x_{ink}) \]
  - No color repeated in the same column (sets of negative binary clauses);
    \[ \forall jk (\neg x_{1jk} \lor \neg x_{2jk}) \land (\neg x_{1jk} \lor \neg x_{3jk}) \ldots (\neg x_{1jk} \lor \neg x_{njk}) \ldots (\neg x_{1jk} \lor \neg x_{(n-1)jk} \lor \neg x_{njk}) \]
Logics in AI: Similar slide to the one for planning

Propositional Definite Clause Logics

Propositional Logics

First-Order Logics

Description Logics

Ontologies

Semantic Web

Information Extraction

Semantics and Proof Theory

Production Systems

Cognitive Architectures

Video Games

Summarization

Tutoring Systems

Satisfiability Testing (SAT)

Hardware Verification

Product Configuration

You will know a little some applications

CPSC 422, Lecture 21
Relationships between different Logics (better with colors)

First Order Logic
\[ \forall X \exists Y p(X, Y) \iff \neg q(Y) \]
- \[ p(a_1, a_2) \]
- \[ \neg q(a_5) \]

Propositional Logic
\[ 7(p \lor q) \rightarrow (r \land s \land t) \]
- \[ p, r \]

Datalog
\[ p(X) \leftarrow q(X) \land r(X, Y) \]
\[ r(x, y) \leftarrow s(y) \]
\[ s(a_1), q(a_2) \]

PDCCL
\[ p \leftarrow s n f \]
\[ r \leftarrow s n g n p \]
Lecture Overview

• Finish SAT (example)
• **First Order Logics**
  • Language and Semantics
  • Inference
Representation and Reasoning in Complex domains (from 322)

- In complex domains expressing knowledge with propositions can be quite limiting.

\[
\begin{align*}
\text{up}(s_2) \\
\text{up}(s_3) \\
\text{ok}(cb_1) \\
\text{ok}(cb_2) \\
\text{live}(w_1) \\
\text{connected}(w_1, w_2)
\end{align*}
\]

- It is often natural to consider individuals and their properties.

\[
\begin{align*}
\text{up}(s_2) \\
\text{up}(s_3) \\
\text{ok}(cb_1) \\
\text{ok}(cb_2) \\
\text{live}(w_1) \\
\text{connected}(w_1, w_2)
\end{align*}
\]

There is no notion that

\[
\begin{align*}
\text{up}(s_2) \\
\text{up}(s_3)
\end{align*}
\]

are about the same property.

\[
\begin{align*}
\text{live}(w_1) \\
\text{connected}(w_1, w_2)
\end{align*}
\]

are about the same individual.
By breaking propositions into relations applied to individuals?

• Express **knowledge** that **holds for set of individuals** (by introducing **variables**)

\[
live(W) \leftarrow connected\_to(W,W1) \land live(W1) \land wire(W) \land wire(W1).
\]

• We can **ask generic queries** (i.e., containing **variables**)

\[
? connected\_to(W, w_1)
\]
“Full” First Order Logics (FOL)

LIKE DATALOG: Whereas propositional logic assumes the world contains facts, FOL (like natural language) assumes the world contains

- **Objects**: people, houses, numbers, colors, baseball games, wars, …
- **Relations**: red, round, prime, brother of, bigger than, part of, comes between, …
- **Functions**: father of, best friend, one more than, plus, …

FURTHERMORE WE HAVE

- **More Logical Operators**: …
- **Equality**: coreference (two terms refer to the same object)
- **Quantifiers**
  - ✓ Statements about unknown objects
  - ✓ Statements about classes of objects
# Syntax of FOL

<table>
<thead>
<tr>
<th>Category</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>Constants</td>
<td>KingJohn, 2, ...</td>
</tr>
<tr>
<td>Predicates</td>
<td>Brother, &gt;, ...</td>
</tr>
<tr>
<td>Functions</td>
<td>Sqrt, LeftLegOf, ...</td>
</tr>
<tr>
<td>Variables</td>
<td>x, y, a, b, ...</td>
</tr>
<tr>
<td>Connectives</td>
<td>( \neg, \Rightarrow, &amp;, \lor, \Leftrightarrow )</td>
</tr>
<tr>
<td>Equality</td>
<td>=</td>
</tr>
<tr>
<td>Quantifiers</td>
<td>( \forall, \exists )</td>
</tr>
</tbody>
</table>
Atomic sentences

**Term** is a *function* \((term_1, \ldots, term_n)\) or constant or variable

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

E.g.,

- \(\text{predicate}(\text{constant}, \text{constant})\)
- \(\text{Brother}(\text{KingJohn}, \text{RichardTheLionheart})\)
- \(\text{predicate}(\text{function}(\text{function}(\text{constant})), \text{function}(\text{function}(\text{constant})))\)
- \(> (\text{Length}(\text{LeftLegOf}(\text{Richard})), \text{Length}(\text{LeftLegOf}(\text{KingJohn})))\)
Complex sentences

Complex sentences are made from atomic sentences using connectives

\[ \neg S, \ S_1 \land S_2, \ S_1 \lor S_2, \ S_1 \Rightarrow S_2, \ S_1 \Leftrightarrow S_2, \]

E.g.

\[ \text{Sibling(KingJohn, Richard)} \Rightarrow \text{Sibling(Richard, KingJohn)} \]

\[ \forall x \ P(x) \text{ is true in an interpretation } I \iff \text{ P is true with } x \]
\[ \text{being each possible object in } I \]

\[ \exists x \ P(x) \text{ is true in an interpretation } I \iff \text{ P is true with } x \]
\[ \text{being some possible object in } I \]
Truth in first-order logic

Like in Prop. Logic a sentence is true with respect to an interpretation.

\[ \neg A \land (B \implies C), \]

\[ \begin{array}{ccc}
A & B & C \\
T & T & F \\
T & F & F \\
F & T & T \\
\end{array} \]

In FOL interpretations are **much more complex** but still the same idea: possible configuration of the world.

- **2 objects** \( \triangle \square \)
- **2 constant symbols** \( \{c_1, c_2\} \) \( \rightarrow \) \( \triangle \) \( \square \)
- **1 unary predicate** \( P \) \( \rightarrow \) \{\( \triangle \)\}
- **1 binary predicate** \( Q \) \( \rightarrow \) \{\{\( \triangle \), \( \triangle \)\}, \{\( \triangle \), \( \square \)\}\}

Is \( \forall x \ P(x) \) TRUE?

A. yes
B. no
Truth in first-order logic

Like in Prop. Logic, a sentence is true with respect to an interpretation:

\[ \neg A \land B \Rightarrow C, \]

\[ \begin{array}{ccc}
A & B & C \\
T & F & F \\
T & T & T \\
\end{array} \]

In FOL interpretations are much more complex but still the same idea: possible configuration of the world.

- **Constants** $\{c_1, c_2\} \rightarrow \{\Delta, \Box\}$
- **Predicates** $P \rightarrow \{\Delta\}$
- **Binary Predicates** $Q \rightarrow \{\{\Delta, \Delta\}, \{\Box, \Box\}\}$

Is $\forall x \ P(x)$ true? NO YES!
Interpretations for FOL: Example

- Binary relations: 2
- Unary relations: 3
- Functions: 1

5 objects

Constant symbols: 5

CPSC 422, Lecture 22
Since we have a one to one mapping between symbols and object we can use symbols to refer to objects

• \{R, J, RLL, JLL, C\}

Property Predicates
• Person = \{R, J\}
• Crown = \{C\}
• King = \{J\}

Relational Predicates
• Brother = \{<R,J>, <J,R>\}
• OnHead = \{<C,J>\}

Functions
• LeftLeg = \{<R, RLL>, <J, JLL>\}
How many Interpretations with….

5 Objects and 5 symbols
• \{R, J, RLL, JLL, C\}

3 Property Predicates (Unary Relations)
• Person
• Crown
• King

2 Relational Predicates
• Brother
• OnHead

1 Function
• LeftLeg

\[ 5! \times (2^5)^3 \times (2^{25})^2 \times 5^5 \]
To summarize: Truth in first-order logic

- Sentences are true with respect to an **interpretation**
- World contains objects (**domain elements**) 
- Interpretation specifies referents for
  - constant symbols → objects
  - predicate symbols → relations
  - function symbols → functional relations

- An atomic sentence \( \text{predicate}(\text{term}_1, \ldots, \text{term}_n) \) is true iff the **objects** referred to by \( \text{term}_1, \ldots, \text{term}_n \) are in the **relation** referred to by \( \text{predicate} \)
Quantifiers

Allows us to express

- **Properties of collections of objects** instead of enumerating objects by name
- **Properties of an unspecified object**

Universal: “for all” $\forall$

Existential: “there exists” $\exists$
Universal quantification

\[ \forall \langle \text{variables} \rangle \langle \text{sentence} \rangle \]

Everyone at UBC is smart:
\[ \forall x \text{ At}(x, \text{UBC}) \implies \text{Smart}(x) \]

\[ \forall x \ P \text{ is true in an interpretation } I \text{ iff } P \text{ is true with } x \text{ being each possible object in } I \]

Equivalent to the conjunction of instantiations of \( P \)

\[ \text{At}(\text{KingJohn}, \text{UBC}) \implies \text{Smart}(\text{KingJohn}) \]
\[ \wedge \text{At}(\text{Richard}, \text{UBC}) \implies \text{Smart}(\text{Richard}) \]
\[ \wedge \text{At}(\text{Ralphie}, \text{UBC}) \implies \text{Smart}(\text{Ralphie}) \]
\[ \wedge \ldots \]
Existential quantification

\[ \exists \text{variables} \quad \exists \text{sentence} \]

Someone at UBC is smart:

\[ \exists x \, \text{At}(x, \text{UBC}) \land \text{Smart}(x) \]

\[ \exists x \quad P \text{ is true in an interpretation } I \text{ iff } P \text{ is true with } x \text{ being some possible object in } I \]

Equivalent to the disjunction of instantiations of \( P \)

\[ \text{At}(\text{KingJohn}, \text{UBC}) \land \text{Smart}(\text{KingJohn}) \]
\[ \lor \quad \text{At}(\text{Richard}, \text{UBC}) \land \text{Smart}(\text{Richard}) \]
\[ \lor \quad \text{At}(\text{Ralphie}, \text{UBC}) \land \text{Smart}(\text{Ralphie}) \]
\[ \lor \quad \ldots \]
Properties of quantifiers

\( \exists x \ \forall y \) 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\((x,\text{IceCream})\) \(\rightarrow\) \(\neg \exists x \ \neg \)Likes\((x,\text{IceCream})\)

\( \exists x \) Likes\((x,\text{Broccoli})\) \(\rightarrow\) \(\forall x \ \neg \)Likes\((x,\text{Broccoli})\)
Lecture Overview

- Finish SAT (example)
- First Order Logics
  - Language and Semantics
  - Inference
FOL: Inference

Resolution Procedure can be generalized to FOL

• Every formula can be rewritten in logically equivalent CNF
  • Additional rewriting rules for quantifiers

• Similar Resolution step, but variables need to be unified (like in DATALOG)

\[
\begin{align*}
\{ & \neg \text{In}(x, y) \lor \neg \text{Charged}(x) \\
\neg \text{In}(z, v) \lor \text{Connected}(z) \\
\rightarrow & \neg \text{Charged}(x) \lor \text{Connected}(x) \}
\end{align*}
\]

\[ \Theta = \{ z/x, v/y \} \]
NLP Practical Goal for FOL: the ultimate Web question-answering system?

Map NL queries into FOPC so that answers can be effectively computed

What African countries are not on the Mediterranean Sea?

\[ \exists c \text{ Country}(c) \land \neg \text{Borders}(c, \text{Med.Sea}) \land \text{In}(c, \text{Africa}) \]

- Was 2007 the first El Nino year after 2001?

\[ \text{ElNino}(2007) \land \neg \exists y \text{ Year}(y) \land \text{After}(y, 2001) \land \text{Before}(y, 2007) \land \text{ElNino}(y) \]
Learning Goals for today’s class

You can:

• Explain differences between Proposition Logic and First Order Logic
• Compute number of interpretations for FOL
• Explain the meaning of quantifiers
• Describe application of FOL to NLP: Web question answering
Next class Fri

- Ontologies (e.g., Wordnet, Probase), Description Logics…
- Midterm will be likely returned next week

Assignment-3 will be out soon (tonight or tomorrow at the latest)