First-Order Logic

Outline

- Why FOL?
- Syntax and semantics of FOL
- Using FOL
- Wumpus world in FOL

Pros and cons of propositional logic

- ☺ Propositional logic is declarative
  - Contrast with procedural approach (Programming languages C++, Java)
  - knowledge and inference are separate, inference is entirely domain-independent
- ☺ ☺ 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

3

First-order logic

- Whereas propositional logic assumes the world contains facts,
  - PL : facts hold or do not hold.
- First-order logic (like natural language) assumes the world contains
  - Objects: people, houses, numbers, colors, squares, pits, wumpus, …
  - Relations: red, round, prime, brother of, bigger than, adjacent to, …
  - Functions: father of, location of (wumpus), one more than, plus, …
- FOL : objects with relations between them that hold or do not hold

4

Syntax of FOL: Basic elements

• Constants	John, 2, ISU,...
• Predicates	Brother, Teacher, Breezy, >,...
• Functions	Sqrt, LeftLegOf,...
• Variables	x, y, a, b,...
• Connectives	$\neg$, $\Rightarrow$, $\wedge$, $\vee$, $\Leftrightarrow$
• Equality	=
• Quantifiers	$\forall$, $\exists$

5

Atomic sentences

Atomic sentence = *predicate* ($term_1$,...,$term_n$)
or $term_1$ = $term_2$

Term = *function* ($term_1$,...,$term_n$)
or *constant* or *variable*

(A term refers to an object)

- E.g., *Brother(KingJohn,RichardTheLionheart)*

Length(LeftLegOf(Richard))=Length(LeftLegOf(KingJohn))

6

Complex sentences

- Complex sentences are made from atomic sentences using connectives

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

E.g. *Sibling(KingJohn,Richard)* $\Rightarrow$ *Sibling(Richard,KingJohn)*

$>(1,2) \vee \leq (1,2)$

$>(1,2) \wedge \neg >(1,2)$

7

Complex sentences

- Universal quantification

  ∀<*variables*> <*sentence*>

  Everyone at ISU is smart:

  $\forall$x At(x,ISU) $\Rightarrow$ Smart(x)

- Existential quantification

  ∃<*variables*> <*sentence*>

  Someone at ISU is smart:

  $\exists x$ At(x, ISU) $\wedge$ Smart(x)

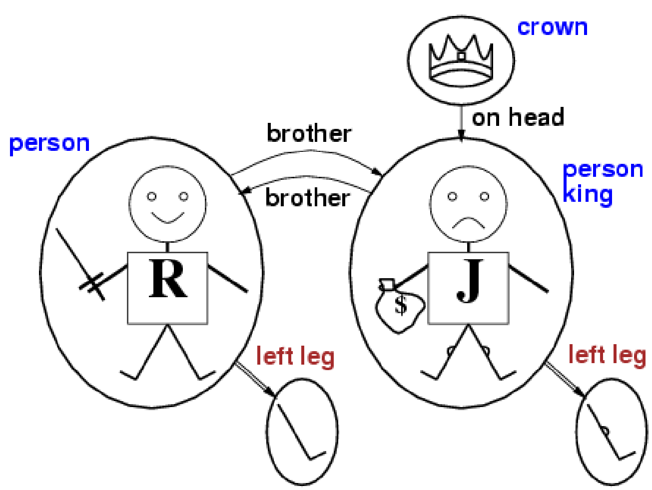
8

Truth in first-order logic

- Sentences are true with respect to a model
- Model contains objects (called domain elements) and an interpretation of symbols
- Interpretation specifies referents for
  constant symbol → object in domain D
  predicate symbol → relation
  function symbol → functional relation
- Each predicate symbol of arity k is mapped to a relation {<object1, …, objectk>}, a set of k-tuples over D^k which are true or, equivalently, a function from D^k to {*true*,*false*}
- Each function symbol of arity k is mapped to a function from D^k to D+1 (domain + an invisible object)

9

Models for FOL: Example



10

Models for FOL: Example

- The intended interpretation
  - Constant Richard → Richard the Lionheart
  - Constant John → King John
  - Predicate Brother → the brotherhood relation
    {< Richard the Lionheart, King John>, <King John, Richard the Lionheart>}
  - Predicates OnHead, Person, King, Crown …
  - Function Leftleg ->
    <Richard the Lionheart> → Richard's left leg
    < King John> → John's left leg

- Another interpretation
  - Constant Richard → the crown
  - Constant John → King John's left leg
  - Predicate Brother →
    {< Richard the Lionheart, the crown>}
  - …

11

Truth in first-order logic

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

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

- The semantics of sentences formed with logical connectives is identical to that in PL

12

Quantifiers

- Allows us to express properties of collections of objects instead of enumerating objects by name
- Universal: "for all" $\forall$
- Existential: "there exists" $\exists$

13

Universal quantification

$\forall$<*variables*> <*sentence*>

Everyone at ISU is smart:

$\forall$x At(x,ISU) $\Rightarrow$ 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*

At(KingJohn,ISU) $\Rightarrow$ Smart(KingJohn)
$\wedge$ At(Richard, ISU) $\Rightarrow$ Smart(Richard)
$\wedge$ At(ISU,ISU) $\Rightarrow$ Smart(ISU)
$\wedge$ ...

14

A 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, ISU) $\wedge$ Smart(x)

  means “Everyone is at ISU and everyone is smart”

15

Existential quantification

$\exists$<*variables*> <*sentence*>

Someone at ISU is smart:
  $\exists x$ At(x, ISU) $\wedge$ 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*

  At(KingJohn,ISU) $\wedge$ Smart(KingJohn)
  $\vee$ At(Richard,ISU) $\wedge$ Smart(Richard)
  $\vee$ At(ISU, ISU) $\wedge$ Smart(ISU)
  $\vee$ ...

16

Another common mistake to avoid

- Typically, $\wedge$ is the natural connective with $\exists$
- Common mistake: using $\Rightarrow$ as the main connective with $\exists$:

  $$\exists x\ At(x, ISU) \Rightarrow Smart(x)$$

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

17

Properties of quantifiers

$\forall x\ \forall y$ is the same as $\forall y\ \forall x$, $\forall y,x$
$\exists x\ \exists y$ is the same as $\exists y\ \exists x$, $\exists y,x$

$\exists x\ \forall y$ 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) $\neg\exists x\ \neg$Likes(x,IceCream)
  $\exists x$ Likes(x,Broccoli) $\neg\forall x\ \neg$Likes(x,Broccoli)

18

Using FOL

The kinship domain:

Predicates *Parent*, *Brother*, *Sibling*, *Child*, *Son*, *Wife*, *Cousin*, …

- Brothers are siblings
  $\forall$x,y *Brother(x,y)* $\Rightarrow$ *Sibling(x,y)*
- One's mother is one's female parent
  $\forall$m,c *Mother(c)* = m $\Leftrightarrow$ *(Female(m)* $\wedge$ *Parent(m,c))*
  (use Mother(m,c)?)
- "Sibling" is symmetric

19

Using FOL

The kinship domain:

- Brothers are siblings
  $\forall$x,y *Brother(x,y)* $\Rightarrow$ *Sibling(x,y)*
- One's mother is one's female parent
  $\forall$m,c *Mother(c)* = m $\Leftrightarrow$ *(Female(m)* $\wedge$ *Parent(m,c))*
- "Sibling" is symmetric
  $\forall$x,y *Sibling(x,y)* $\Leftrightarrow$ *Sibling(y,x)*
- A first cousin is a child of a parent's sibling

20

Using FOL

The kinship domain:

- Brothers are siblings
  $\forall$x,y *Brother(x,y)* $\Rightarrow$ *Sibling(x,y)*
- One's mother is one's female parent
  $\forall$m,c *Mother(c)* = m $\Leftrightarrow$ *(Female(m)* $\wedge$ *Parent(m,c))*
- "Sibling" is symmetric
  $\forall$x,y *Sibling(x,y)* $\Leftrightarrow$ *Sibling(y,x)*
- A first cousin is a child of a parent's sibling
  $\forall$x,y *FirstCousin(x,y)* $\Leftrightarrow$ $\exists$p,ps Parent(p,x) $\wedge$ Sibling(ps,p) $\wedge$ Parent(ps,y)
- Definition of full *Sibling* in terms of *Parent*:

21

Using FOL

The kinship domain:

- Brothers are siblings
  $\forall$x,y *Brother(x,y)* $\Rightarrow$ *Sibling(x,y)*
- One's mother is one's female parent
  $\forall$m,c *Mother(c)* = m $\Leftrightarrow$ *(Female(m)* $\wedge$ *Parent(m,c))*
- "Sibling" is symmetric
  $\forall$x,y *Sibling(x,y)* $\Leftrightarrow$ *Sibling(y,x)*
- A first cousin is a child of a parent's sibling
  $\forall$x,y *FirstCousin(x,y)* $\Leftrightarrow$ $\exists$p,ps Parent(p,x) $\wedge$ Sibling(ps,p) $\wedge$ Parent(ps,y)
- Definition of full *Sibling* in terms of *Parent*:
  $\forall$*x,y FullSibling(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)]

22

Using FOL - The set domain

The set theory can be built up from a tiny kernel of axioms

Constant {}, unary predicate *Set*

Syntactic sugar: an extension to the standard syntax to make sentences easier to read

- Binary functions: $\{x|s\}$, $s_1 \cap s_2$, $s_1 \cup s_2$
- Binary predicates: $x \in s$, $s_1 \subseteq s_2$

- $\forall s\ Set(s) \Leftrightarrow (s = \{\}) \vee (\exists x,s_2\ Set(s_2) \wedge [s = \{x|s_2\}])$
- $\neg\exists x,s \quad \{x|s\} = \{\}$
- $\forall x,s \quad x \in s \Leftrightarrow s = \{x|s\}$
- $\forall x,s \quad x \in s \Leftrightarrow [\ \exists y,s_2\ (s = \{y|s_2\} \wedge (x = y \vee x \in s_2))]$
- $\forall s_1,s_2 \quad s_1 \subseteq s_2 \Leftrightarrow (\forall x\ \ x \in s_1 \Rightarrow x \in s_2)$
- $\forall s_1,s_2 \quad (s_1 = s_2) \Leftrightarrow (s_1 \subseteq s_2 \wedge s_2 \subseteq s_1)$
- $\forall x,s_1,s_2 \quad x \in (s_1 \cap s_2) \Leftrightarrow (x \in s_1 \wedge x \in s_2)$
- $\forall x,s_1,s_2 \quad x \in (s_1 \cup s_2) \Leftrightarrow (x \in s_1 \vee x \in s_2)$

23

Example Knowledge for wumpus world

- Unary Predicate *Pit()*, *Breezy()*, *Smelly()*
- constant *Wumpus* or *Wumpus()*?
- function *Home(Wumpus)* to name the one square with wumpus, or In(Wumpus, s)?
- Squares are breezy near a pit:

24

Example Knowledge for wumpus world

- Unary Predicate *Pit()*, *Breezy()*, *Smelly()*
- Squares are breezy near a pit:
  - Diagnostic rule---infer cause from effect
    $\forall s\ Breezy(s) \Leftrightarrow \exists\ r\ Adjacent(r,s) \wedge Pit(r)$
  - Causal rule---infer effect from cause
    $\forall r\ Pit(r) \Rightarrow [\forall s\ Adjacent(r,s) \Rightarrow Breezy(s)]$
    (not complete!)
    $\forall s\ [\forall r\ Adjacent(r,s) \Rightarrow \neg Pit(r)] \Rightarrow \neg\ Breezy(s)$

25

Example Knowledge for wumpus world

- Actions:
  Turn(Right), Turn(Left), Forward, Shoot, Grab
- Properties of locations:
  $\forall s,t\ At(Agent,s,t) \wedge Smell(t) \Rightarrow Smelly(s)$
  $\forall s,t\ At(Agent,s,t) \wedge Breeze(t) \Rightarrow Breezy(s)$
- Reflex behavior
  - $\forall t\ Glitter(t) \Rightarrow BestAction(Grab,t)$

26

Summary

- First-order logic:
  - objects and relations are semantic primitives
  - syntax: constants, functions, predicates, equality, quantifiers.
- Increased expressive power: sufficient to define wumpus world

27