Declarative language

Before building system

before there can be learning, reasoning, planning, explanation ...

need to be able to express knowledge

Want a precise declarative language

- declarative: believe P = hold P to be <u>true</u>

  cannot believe P without some sense of what it would mean for the world to satisfy P

- precise: need to know exactly
  - what strings of symbols count as sentences
  - what it means for a sentence to be true

    (but without having to specify which ones are true)

What does it mean to have a language?

- syntax
- semantics
- pragmatics

Here: language of first-order logic

again: not the only choice

Alphabet

Logical symbols:

- Punctuation: (,), .
- Connectives: $\neg$, $\wedge$, $\vee$, $\supset$, $\equiv$, $\forall$, $\exists$, $=$
- Variables: x, x_1, x_2, ..., x', x'', ..., y, ..., z, ...

    Fixed meaning and use

    like keywords in a programming language

Non-logical symbols

- Predicate symbols (like Dog)
- Function symbols (like bestFriendOf)

    Domain-dependent meaning and use

    like identifiers in a programming language

Have <u>arity</u>: number of arguments

arity 0 predicates: propositional symbols

arity 0 functions: constant symbols

Assume infinite supply of every arity

Note: not treating $=$ as a predicate

Grammar

Expressions: terms and formulas (wffs)

Terms

1. Every variable is a term.
2. If $t_1, t_2, ..., t_n$ are terms and f is a function of arity n, then $f(t_1, t_2, ..., t_n)$ is a term.

Atomic wffs

1. If $t_1, t_2, ..., t_n$ are terms and P is a predicate of arity n, then $P(t_1, t_2, ..., t_n)$ is an atomic wff.
2. If t_1 and t_2 are terms, then $(t_1 = t_2)$ is an atomic wff.

Wffs

1. Every atomic wff is a wff
2. If α and β are wffs, and v is a variable, then $\neg\alpha$, $(\alpha \wedge \beta)$, $(\alpha \vee \beta)$, $\exists v.\alpha$, $\forall v.\alpha$ are wffs.

The propositional subset:

No terms

Atomic wffs: only predicates of 0-arity

No variables and no quantifiers

$$(p \wedge \neg(q \vee r))$$

Notation

Occasionally add or omit (,), .

Use [,] and {, } also.

Abbreviations:

$(\alpha \supset \beta)$ for $(\neg\alpha \vee \beta)$

$(\alpha \equiv \beta)$ for $((\alpha \supset \beta) \wedge (\beta \supset \alpha))$

Non-logical symbols:

Predicates: Person, Happy, OlderThan

Functions: fatherOf, successor, johnSmith

Lexical scope for variables

$P(x) \wedge \exists x[P(x) \vee Q(x)]$

free bound occurrences of variables

Sentence: wff with no free variables (closed)

Substitution: $\alpha[v/t]$ means α with all free occurrences of v replaced by term t (also α^{v}_{t})..

Semantics

How to interpret sentences?

- what do sentences claim about the world?
- what does believing one amount to?

Without answers, cannot use sentences to represent knowledge

Problem:

cannot fully specify interpretation of sentences because non-logical symbols reach outside the language

So:

make clear dependence of interpretation on
non-logical symbols

Logical interpretation:

specification of how to understand predicate and function symbols

Can be complex!

DemocraticCountry,
IsABetterJudgeOfCharacterThan,
favouriteIceCreamFlavourOf,
puddleOfWater27

Simple case

There are objects

some satisfy predicate P; some do not

Each interpretation settles extension of P

borderline cases ruled in separate interpretations

Each interpretation assigns to function f a mapping from objects to objects

functions always well-defined and single-valued

Main assumption:

this is all you need to know about the non-logical symbols to understand which sentences of FOL are true or false

In other words, given a specification of

- what objects there are
- which of them satisfy P
- what mapping is denoted by f

it will be possible to say which sentences of FOL are true and which are not

Interpretations

Two parts: $\boldsymbol{I} = \langle D, \Phi \rangle$

D is the domain of discourse

__can be any set

not just formal / mathematical objects

e.g. people, tables, numbers, sentences, chunks of peanut butter, situations, the universe

Φ is an interpretation mapping

If *P* is a predicate symbol of arity n,

$\Phi(P) \subseteq [D \times D \times ... \times D]$

an n-ary relation over *D*

Can view interpretation of predicates

in terms of characteristic function

$\Phi(P) \in [D \times D \times ... \times D \rightarrow \{0, 1\}]$

If *f* is a function symbol of arity n,

$\Phi(f) \in [D \times D \times ... \times D \rightarrow D]$

an n-ary function over *D*

For constants, $\Phi(c) \in D$

Denotation

In terms of interpretation $\boldsymbol{I}$, terms will denote elements of *D*.

will write element as $\boldsymbol{I}\|t\|$

For terms with variables, denotation depends on the values of variables

will write as $\boldsymbol{I},\mu\|t\|$

where $\mu \in [\mathit{Variables} \rightarrow D]$,
called a variable assignment

Rules of interpretation:

1. $\boldsymbol{I},\mu\ \|v\| = \mu(v)$.
2. $\boldsymbol{I},\mu\ \| f(t_1, t_2, ..., t_n) \| = H(d_1, d_2, ..., d_n)$

   where $H = \Phi(f)$

   and $d_i = \boldsymbol{I},\mu\|t_i\|$, recursively

Satisfaction

In terms of $\boldsymbol{I}$, wffs will be true for some values of the free variables and false for others

will write as $\boldsymbol{I},\mu \models \alpha$ "α is satisfied by $\boldsymbol{I}$ and μ"

where $\mu \in$ [*Variables* $\rightarrow$ *D*], as before

or $\boldsymbol{I} \models \alpha$, when α is a sentence

or $\boldsymbol{I} \models S$, when S is a set of sentences

(*all* sentences in S are true in $\boldsymbol{I}$).

Rules of interpretation:

1. $\boldsymbol{I},\mu \models P(t_1, t_2, ..., t_n)$ iff $\langle d_1, d_2, ..., d_n \rangle \in R$
   where $R = \Phi(P)$
   and $d_i = \boldsymbol{I},\mu \| t_i \|$, as on previous slide
2. $\boldsymbol{I},\mu \models (t_1 = t_2)$ iff $\boldsymbol{I},\mu\| t_1 \|$ is the same as $\boldsymbol{I},\mu\| t_2 \|$
3. $\boldsymbol{I},\mu \models \neg\alpha$ iff $\boldsymbol{I},\mu \not\models \alpha$
4. $\boldsymbol{I},\mu \models (\alpha \wedge \beta)$ iff $\boldsymbol{I},\mu \models \alpha$ and $\boldsymbol{I},\mu \models \beta$
5. $\boldsymbol{I},\mu \models (\alpha \vee \beta)$ iff $\boldsymbol{I},\mu \models \alpha$ or $\boldsymbol{I},\mu \models \beta$
6. $\boldsymbol{I},\mu \models \exists v.\alpha$ iff for some $d \in D$, $\boldsymbol{I},\mu\{d;v\} \models \alpha$
7. $\boldsymbol{I},\mu \models \forall v.\alpha$ iff for all $d \in D$, $\boldsymbol{I},\mu\{d;v\} \models \alpha$
   where $\mu\{d;v\}$ is just like μ, except on v, where $\mu(v)=d$.

For propositional subset:

$\boldsymbol{I} \models p$ iff $\Phi(p) = 1$ and the rest as above

Logical consequence

Semantic rules of interpretation tell us how to understand all wffs in terms of specification for non-logical symbols.

But some connections among sentences are independent of non-logical symbols involved.

e.g. If α is true under $\boldsymbol{I}$, then so is $\neg(\beta \wedge \neg\alpha)$, no matter what $\boldsymbol{I}$ is, why α is true, what β is, ...

a function of logical symbols only

S entails α or α is a logical consequence of *S*:

$S \models \alpha$ iff for every $\boldsymbol{I}$, if $\boldsymbol{I} \models S$ then $\boldsymbol{I} \models \alpha$.

In other words: for no $\boldsymbol{I}$, $\boldsymbol{I} \models S \cup \{\neg\alpha\}$.

Say that $S \cup \{\neg\alpha\}$ is unsatisfiable

Special case: *S* is empty

$\models \alpha$ iff for every $\boldsymbol{I}$, $\boldsymbol{I} \models \alpha$. Say α is valid.

Note: $\{\alpha_1, \alpha_2, ..., \alpha_n\} \models \alpha$ iff $\models (\alpha_1 \wedge \alpha_2 \wedge ... \wedge \alpha_n) \supset \alpha$

finite entailment reduces to validity

Why do we care?

We do not have access to user-intended interpretation of non-logical symbols

But, with entailment, we know that if *S* is true in the intended interpretation, then so is α.

If the user's view has the world satisfying *S,* then it must also satisfy α.

There may be other sentences true also; but α is logically guaranteed.

So what about:

Dog(fido) => Mammal(fido) ??

Not entailment!

There are logical interpretations where

$\Phi(\text{Dog}) \not\subset \Phi(\text{Mammal})$

> Key idea of KR:
>
> include such connections explicitly in *S*
>
> $\forall x[\text{Dog}(x) \supset \text{Mammal}(x)]$
>
> Get: $S \cup \{\text{Dog(fido)}\} \models \text{Mammal(fido)}$

The rest is just the details...

Knowledge Bases

KB is set of sentences

explicit statement of sentences believed (including assumed connections among non-logical symbols)

KB $\models \alpha$ — α is a further consequence of what is believed

- explicit knowledge: KB
- implicit knowledge: $\{ \alpha \mid \text{KB} \models \alpha \}$

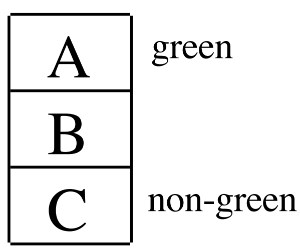
Often non trivial: explicit ? implicit

Example:

Three blocks stacked.

Top one is green.

Bottom one is not green.



Is there a green block directly on top of a non-green block?

A formalization

S = {On(a,b), On(b,c), Green(a), ¬Green(c)}

all that is required

$\alpha = \exists x \exists y[\text{Green}(x) \wedge \neg\text{Green}(y) \wedge \text{On}(x,y)]$

<u>Claim:</u> $S \models \alpha$

<u>Proof:</u>

Let $\boldsymbol{I}$ be any interpretation such that $\boldsymbol{I} \models S$.

Case 1: $\boldsymbol{I} \models$ Green(b).

$\therefore\ \boldsymbol{I} \models \text{Green(b)} \wedge \neg\text{Green(c)} \wedge \text{On(b,c)}$.

$\therefore\ \boldsymbol{I} \models \alpha$

Case 2: $\boldsymbol{I} \not\models$ Green(b).

$\therefore\ \boldsymbol{I} \models \neg\text{Green(b)}$

$\therefore\ \boldsymbol{I} \models \text{Green(a)} \wedge \neg\text{Green(b)} \wedge \text{On(a,b)}$.

$\therefore\ \boldsymbol{I} \models \alpha$

Either way, for any $\boldsymbol{I}$, if $\boldsymbol{I} \models S$ then $\boldsymbol{I} \models \alpha$.

So $S \models \alpha$. QED

Knowledge-based system

Start with (large) KB representing what is explicitly known

e.g. what the system has been told

Want to influence behaviour based on what is <u>implicit</u> in the KB (or as close as possible)

Requires reasoning

<u>deductive inference</u>:

process of calculating entailments of KB

i.e given KB and any α, determine if KB $\models \alpha$

Process is <u>sound</u> if whenever it produces α, then KB $\models \alpha$

does not allow for plausible assumptions that may be true in intended interpretation

Process is <u>complete</u> if whenever KB $\models \alpha$, it produces α

does not allow for process to miss some α or be unable to determine the status of α