Artificial Intelligence

5. Knowledge Representation using Formal Logic

Prof. Bojana Dalbelo Bašić
Assoc. Prof. Jan Šnajder

University of Zagreb
Faculty of Electrical Engineering and Computing

Academic Year 2019/2020

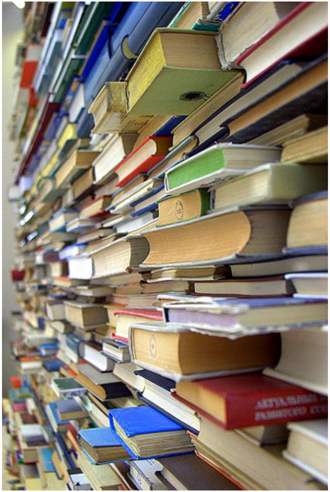
 1.4

Outline

Outline

Motivation

- **Knowledge representation** is the central problem of artificial intelligence
- Solving many real-world problems requires **huge amounts of knowledge**, even when restricted to narrow domains
- Another important task is **inference**: how to use the represented knowledge to derive new knowledge?
- Inference is required for decision making, planning, language understanding, theorem proving, etc.



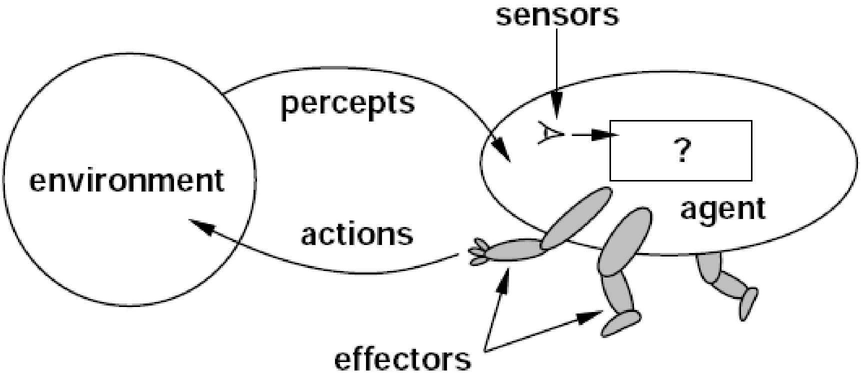
Motivation: Virtual assistants

Learning fact	**User:**	If it rains hard, then the roads get slick.
	Hal:	I will remember that one follows the other. Caught in a rainstorm it's time to seek shelter.
Learning fact	**User:**	If the roads get slick, then drivers need to slow down.
	Hal:	I see the relationship. It would be nice to get rain when we need it.
Learning fact	**User:**	If drivers need to slow down, then traffic moves more slowly.
	Hal:	Yes, I get that clearly.
Deductive reasoning	**User:**	Ok, so what happens when it rains hard?
	Hal:	Traffic moves more slowly because drivers need to slow down.
Chat	**User:**	You learn well.
	Hal:	I am constantly learning from what you teach me.
Chat	**User:**	It's been nice talking with you, but I have to go now
	Hal:	No, don't go! I like talking to you. I think about you when you are not here.
Chat	**User:**	That's nice to know, but I can't stay, good bye.
	Hal:	See you later.

"Ultra Hal digital secretary and companion", winner of the Loebner Prize 2007

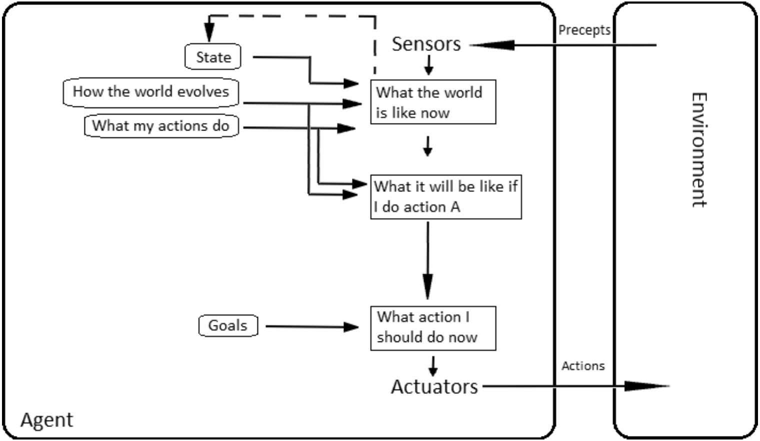
`https://www.zabaware.com/ultrahal/`

Motivation: Intelligent agent



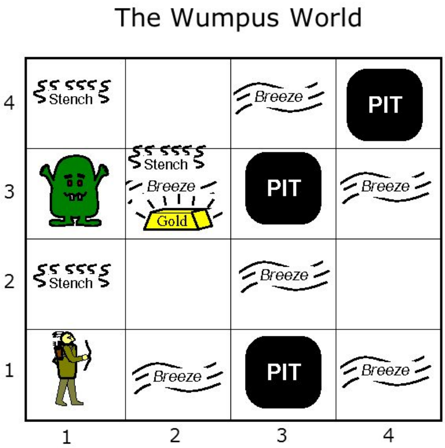
From: Russel & Norving. Artificial Intelligence: A Modern Approach.

Motivation: Intelligent agent



From: Russel & Norving. Artificial Intelligence: A Modern Approach.

Motivation: The Wumpus World



Percepts (facts):

$\neg B_{1,1}$
$\neg B_{1,2}$
$B_{2,1}$
$\neg S_{1,1}$
$S_{1,2}$
$\neg S_{2,1}$
$\neg P_{1,1}$
$\neg W_{1,1}$
$\neg G_{1,1}$

Knowledge (rules):

$\vdots$

$B_{2,1} \leftrightarrow (P_{1,1} \vee P_{2,2} \vee P_{3,1})$
$S_{1,2} \leftrightarrow (W_{1,1} \vee W_{2,2} \vee W_{1,3})$

$\vdots$

Example: Customs officers and diplomats

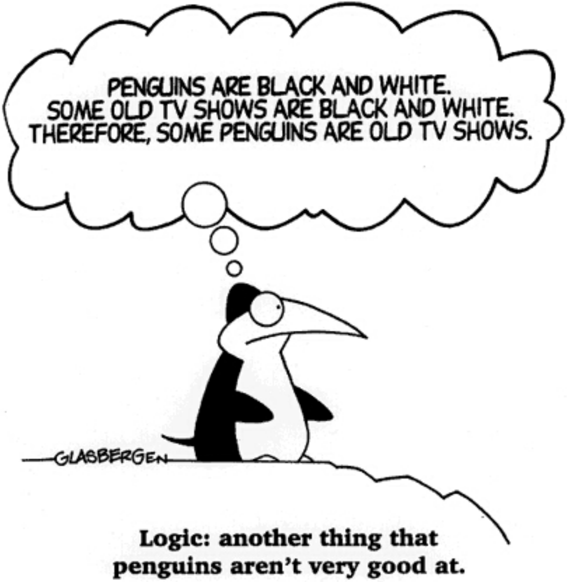
Premises

Customs officers searched everyone who entered the country but wasn't a diplomat. Some smugglers who entered the country were searched only by smugglers. No smuggler is a diplomat.

Conclusion

Some customs officers are smugglers.

Outline


PENGUINS ARE BLACK AND WHITE.
SOME OLD TV SHOWS ARE BLACK AND WHITE.
THEREFORE, SOME PENGUINS ARE OLD TV SHOWS.
GLASBERGEN
Logic: another thing that
penguins aren't very good at.

Symbolism vs. connectionism

- Symbolic logic is in the heart of the **symbolic approach** to artificial intelligence:

Symbolism

Knowledge about the external world can be represented with **symbols**. Inference amounts to symbol **manipulation**. Intelligent reasoning or behaviour amounts to inference.

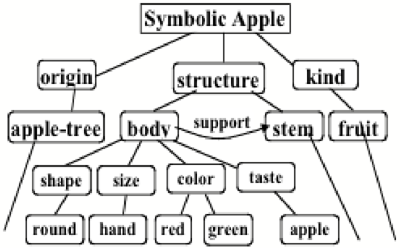
- This is in contrast to **connectionism**:

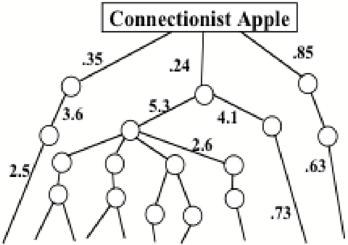
Connectionism

Mental states and behaviour emerges from the **interaction** of a large number of **interconnected** and **simple** processing units. An artificial neural network is a typical example of the connectionist approach to AI.

- We'll stick to the symbolic approach for the time being

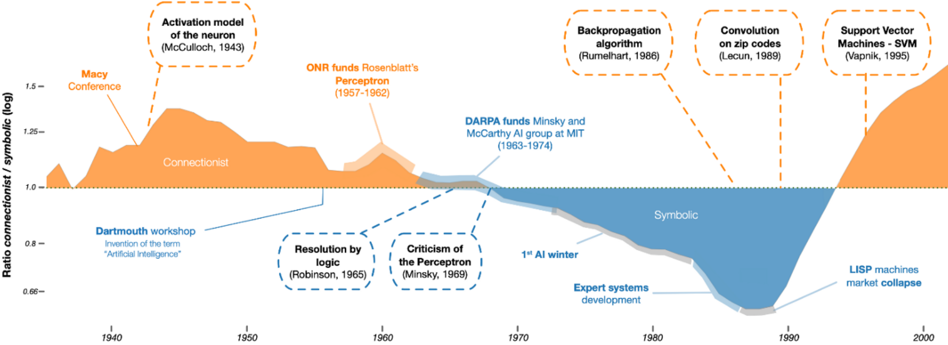
Symbolism vs. connectionism





Minsky, M. (1990). **Logical vs. Analogical or Symbolic vs. Connectionist or Neat vs. Scruffy**. Artificial Intelligence at MIT. Expanding Frontiers, Patrick H. Winston (Ed.).

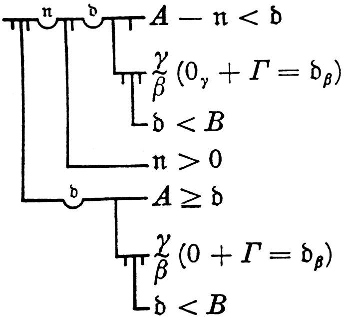
Symbolism vs. connectionism



Cardon, D., Cointet, J. P., & Mazieres, A. (2018). **Neurons spike back: The Invention of Inductive Machines and the Artificial Intelligence Controversy.**

Symbolic logic

Symbolic logic is a branch of mathematics concerned with mathematical concepts expressed using a **formal logic system**. Such a system enables abstract reasoning



Gottlob Frege, *Begriffsschrift*, 1879.

Formal logic system

Every system of formal logic consists of three components:

Components of a formal logic system

1. **Syntax** – describes the structures that constitute the language of logic, i.e., defines formal rules for construction of logic formulae
2. **Semantics** – describes the meaning of language structures. E.g., which structures are true and which are false, or what is the relation between language elements and objects from the external world
3. **Proof theory** – defines the mechanisms of inference, i.e., how new conclusions may be derived from the premises

- (1) and (2) serve to represent the real-world knowledge
- (3) serves for the derivation of new knowledge

Expressivity of logic

- There are many kinds of logics, differing in the three components
- Some are more and some are less **expressive**
- Advantages of high expresivity:
  - allows a more detailed description of the real world
- Downsides of high expresivity:
  - more complex syntax, semantics, and proof theory
  - we can't prove everything $\Rightarrow$ **undecidability**
- A logic is **decidable** if it's possible to check the validity of all its formulae

Trade-off between expressivity and decidability

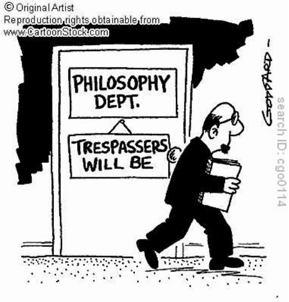
In principle, the more expressive a logic is, the fewer the number of things that can be proved. Very expressive logical systems are not decidable. Systems that are not very expressive are decidable, but cannot express much.

Kinds of logics

- **Propositional logic (sentential logic) $\Rightarrow$ PL**
- **First-order (predicate) logic $\Rightarrow$ FOL**
- Temporal logic
- Description logic
- **Fuzzy logic**
- Modal logic
- Epistemic logic
- ...

Expressivity depends on the **ontological** and **epistemological** commitments

Ontological commitments



Ontology

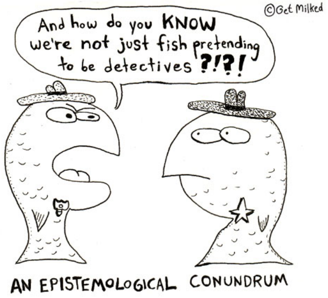
Ontology is a philosophic discipline (branch of metaphysics) concerned with the nature of **existence** and **reality**, the question which entities exist and what are the relations between them.

Ontological commitments

Ontological commitments are our assumptions about what exists in the world. E.g., in **propositional logic** we assume that the world consists of facts that are either true or false, in **temporal logic** we additionally assume that the world consists of ordered points in time, etc.

- **NB**: The term **ontology** has yet another meaning in AI: formal representation of knowledge within a domain (more on this later)

Epistemological commitments



Epistemology

Epistemology is a branch of philosophy concerned with the nature of **knowledge** – its scope, sources, and limitations.

Epistemological commitments

Epistemological commitments define the possible states of knowledge. E.g, in **propositional logic**, each fact can only be either true or false. It cannot be partially true, uncertain, or just be believed in by someone. There are other kinds of logics whose epistemological commitments allow us to express beliefs and different degrees of certainty.

Outline

Syntax of propositional logic (1)

Symbols of propositional logic

(1) Set of **propositional variables** or **atomic formulae**, $V = \{A, B, C, \dots\}$

(2) **Logical operators** or logical connectives:

- $\neg$ (negation)
- $\vee$ (operator *or*)
- $\wedge$ (operator *and*)
- $\rightarrow$ (implication)
- $\leftrightarrow$ (equivalence, biconditional)

(3) Logical constants $True$ and $False$, which denote a a proposition that is always true or false, respectively

(4) Parentheses (and)

Syntax of propositional logic (2)

Well-formed formula (wff)

A **well-formed formula** (wff) or simply **formula** of propositional logic is defined recursively as follows:

(1) An atom is a formula

(2) If F is a formula, then $(\neg F)$ is also a formula

(3) If F and G are formulae, then the following are also formulae:

- $(F \wedge G)$
- $(F \vee G)$
- $(F \rightarrow G)$
- $(F \leftrightarrow G)$

(4) Nothing else is a formula

- Simplifying convention: Parentheses in (2) may be dropped. Parentheses in (3) may be dropped if they are the outer parentheses

Syntax of propositional logic – examples

Examples of atoms:

- $A =$ *"The Earth is round"*
- $B =$ *"Harry Potter attends the Hogwarts School of Witchcraft"*
- $C =$ *"PL is the most powerful knowledge representation scheme"*
- $D =$ *"Minotaur is a mythical creature"*

Examples of formulae:

- C
- $\neg C$
- $((A \vee B) \vee \neg C)$
- $(((B \vee D) \wedge (\neg B \vee C)) \rightarrow (A \vee C))$
- $((C \vee D) \rightarrow (\neg A \leftrightarrow B))$

Semantics of propositional logic

Interpretation

Let F be a well-formed formula of propositional logic. Let $V = \{E_1, E_2, \ldots, E_n\}$ be the set of its propositional variables.

Interpretation $I : V \to \{\top, \bot\}$ of formula F is an assignment of truth values from set $\{\top, \bot\}$ to variables from set V.

Function I assigns to each propositional variable E_i either $I(E_i) = \top$ (true) or $I(E_i) = \bot$ (false), but not both.

- A formula with n atoms has 2^n different interpretations
- Each interpretation I describes one possible **world situation**

Semantic of logic operators

Truth tables

F	G	$\neg F$	$F \wedge G$	$F \vee G$	$F \rightarrow G$	$F \leftrightarrow G$
$\bot$	$\bot$	$\top$	$\bot$	$\bot$	$\top$	$\top$
$\bot$	$\top$	$\top$	$\bot$	$\top$	$\top$	$\bot$
$\top$	$\bot$	$\bot$	$\bot$	$\top$	$\bot$	$\bot$
$\top$	$\top$	$\bot$	$\top$	$\top$	$\top$	$\top$

- Operators have "natural language semantics", except operator $\rightarrow$, whose semantics is less intuitive
- In formula $F \rightarrow G$, formula F is called the **antecedent**, and G is called the **consequent**

Truth value of a formula

Truth value of a formula

Given interpretation I, the truth value of formula F is defined recursively:

$$\begin{aligned}
I(\mathit{True}) &\equiv \top \\
I(\mathit{False}) &\equiv \bot \\
I(\neg F) &\equiv \neg I(F) \\
I(F \vee G) &\equiv I(F) \vee I(G) \\
I(F \wedge G) &\equiv I(F) \wedge I(G)
\end{aligned}$$

- E.g., truth value of $((A \vee B) \wedge C) \wedge (\neg B \vee C)$ in interpretation $I(A) = \bot$, $I(B) = \top$, $I(C) = \top$:

$$\begin{aligned}
&I(((A \vee B) \wedge C) \wedge (\neg B \vee C)) \equiv \\
&I((A \vee B) \wedge C) \wedge I(\neg B \vee C) \equiv \\
&(I(A \vee B) \wedge I(C)) \wedge (I(\neg B) \vee I(C)) \equiv \\
&((I(A) \vee I(B)) \wedge I(C)) \wedge (I(\neg B) \vee I(C)) \equiv \dots
\end{aligned}$$

Model

Model

Interpretation I is a **model** of formula F if and only if F is true in I.
In this case we also say that I **satisfies** formula F.

- A model describes a single **world situation**
- E.g., model of $\neg A \wedge D$ is a situation in which $A = \bot$ (*The Earth is not round*) and $D = \top$ (*Minotaur is a mythical creature*)
- A single formula may have many models, and it then describes many situations at once
  E.g., formula $A \vee D$ has several models. (Which ones?)

Validity, inconsistency, and satisfiability (1)

Valid formula

A formula is **valid** (**tautology**) if and only if it is true in all interpretations.

Inconsistent formula

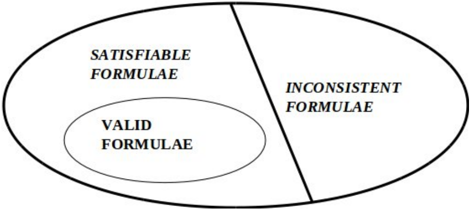
A formula is **inconsistent** (**contradiction**, **unsatisfiable**, **anti-tautology**) if and only if it is false in every interpretation.

Satisfiable formula

A formula is **satisfiable** (**consistent**) if and only if it is true in at least one interpretation.

Validity, inconsistency, and satisfiability (2)

- A formula is consistent iff it is not inconsistent
- A formula is valid iff its negation is inconsistent
- If a formula is valid, then it is satisfiable, but the converse does not hold
- If a formula is not valid, it need not be inconsistent
- If a formula is not inconsistent, then it is consistent, but it need not be valid



Validity, contradiction, and satisfiability – examples

- P – satisfiable
- $P \vee Q$ – satisfiable
- $\neg(P \vee Q)$ – satisfiable
- $\neg P \wedge P$ – inconsistent
- $P \wedge P$ – satisfiable
- $P \rightarrow Q$ – satisfiable
- $P \vee \neg P$ – valid
- $(P \rightarrow Q) \wedge P$ – satisfiable
- $((P \rightarrow Q) \wedge P) \rightarrow Q$ – valid
- $((P \rightarrow Q) \wedge P) \rightarrow \neg Q$ – satisfiable
- $P \leftrightarrow Q$ – satisfiable

Equivalence

Equivalent formulae

Formula F is **equivalent** to formula G, denoted $F \equiv G$, if and only if the truth values of F and G are the same in every interpretation of F and G

- $F \leftrightarrow G$ is valid iff $F \equiv G$ holds

Equivalences of propositional logic (1)

$$
\begin{array}{rlcll}
(1) & \neg\neg F & \equiv & F & \text{– involution} \\
(2) & F \rightarrow G & \equiv & \neg F \vee G & \text{– implication elim.} \\
(3) & F \rightarrow G & \equiv & \neg G \rightarrow \neg F & \text{– contraposition} \\
(4) & F \rightarrow (G \rightarrow H) & \equiv & G \rightarrow (F \rightarrow H) & \\
(5) & F \rightarrow (G \rightarrow H) & \equiv & (F \wedge G) \rightarrow H & \\
(6) & F \leftrightarrow G & \equiv & (F \wedge G) \vee (\neg F \wedge \neg G) & \\
(7) & F \leftrightarrow G & \equiv & (F \rightarrow G) \wedge (G \rightarrow F) & \\
(8) & F \leftrightarrow G & \equiv & (\neg F \vee G) \wedge (\neg G \vee F) & \\
(9) & G \wedge G & \equiv & G & \text{– idempotence} \\
(10) & G \wedge \mathit{True} & \equiv & G & \\
(11) & G \wedge \mathit{False} & \equiv & \mathit{False} & \\
(12) & G \wedge \neg G & \equiv & \mathit{False} & \text{– law of contradiction} \\
(13) & G \vee G & \equiv & G & \text{– factorization} \\
(14) & G \vee \mathit{True} & \equiv & \mathit{True} &
\end{array}
$$

Equivalences of propositional logic (2)

$$
\begin{array}{rlcll}
(15) & G \vee \mathit{False} & \equiv & G & \\
(16) & G \vee \neg G & \equiv & \mathit{True} & \text{– law of excluded middle} \\
(17) & (F \wedge G) \wedge H & \equiv & F \wedge (G \wedge H) & \multirow{2}{*}{\} \text{ associativity}} \\
(18) & (F \vee G) \vee H & \equiv & F \vee (G \vee H) & \\
(19) & F \wedge G & \equiv & G \wedge F & \multirow{2}{*}{\} \text{ commutativity}} \\
(20) & F \vee G & \equiv & G \vee F & \\
(21) & F \vee (G \wedge H) & \equiv & (F \vee G) \wedge (F \vee H) & \multirow{2}{*}{\} \text{ distributivity}} \\
(22) & F \wedge (G \vee H) & \equiv & (F \wedge G) \vee (F \wedge H) & \\
(23) & \neg(F \vee G) & \equiv & \neg F \wedge \neg G & \multirow{2}{*}{\} \text{ de Morgan's laws}} \\
(24) & \neg(F \wedge G) & \equiv & \neg F \vee \neg G & \\
(25) & F \vee (F \wedge G) & \equiv & F & \multirow{2}{*}{\} \text{ absorption laws}} \\
(26) & F \wedge (F \vee G) & \equiv & F & \\
(27) & F \vee (\neg F \wedge G) & \equiv & F \vee G & \\
(28) & F \wedge (\neg F \vee G) & \equiv & F \wedge G &
\end{array}
$$

Logical consequence

- What are the conclusions that follow from given facts (premises)?

Logical consequence

Formula G is a **logical (semantic) consequence** of formulae $F_1, \dots, F_n$ if and only if every interpretation that satisfies $F_1 \wedge \cdots \wedge F_n$ also satisfies G.

Put differently:
Formula G is a logical consequence of formulae $F_1, \dots, F_n$ iff every model of $F_1 \wedge \cdots \wedge F_n$ also is a model of G.

We write $F_1, F_2, \dots, F_n \vDash G$
and read "$F_1, \dots, F_n$ **logically (semantically) entails** G".

Logical consequence – an example

- Let's show that Q is a logical consequence of $P \lor Q$ and $\neg P$, i.e.:

$$P \lor Q, \neg P \ \models \ Q$$

- We start by constructing a truth table:

P	Q	$P \lor Q$	$\neg P$	Q
$\bot$	$\bot$	$\bot$	$\top$	$\bot$
$\bot$	$\top$	$\top$	$\top$	$\top$
$\top$	$\bot$	$\top$	$\bot$	$\bot$
$\top$	$\top$	$\top$	$\bot$	$\top$

- The premises have a single model, which also is a model of Q. Therefore, by definition, Q is the logical consequence of the premises

Logical consequence, validity, and inconsistency

- With $\models F$ we denote that F is a valid formula
- By the definition of logical consequence, a valid formula is logically entailed by any premise (including an empty set of premises)
- For example:
  $Q \ \models \ P \vee \neg P$
  $\models P \vee \neg P$
- Also, by the definition of logical consequence, an inconsistent formula logically entails any formula
- For example:
  $P \wedge \neg P \ \models \ P$
  $P \wedge \neg P \ \models \ Q$

Proving logical consequence (1)

Direct method

Formula G is a logical consequence of $F_1, F_2, \ldots, F_n$ if and only if

$$(F_1 \wedge \cdots \wedge F_n) \rightarrow G$$

is **valid** (tautology). In other words:

$$F_1 \wedge \cdots \wedge F_n \ \vDash \ G$$

if and only if

$$\vDash (F_1 \wedge \cdots \wedge F_n) \rightarrow G$$

- The above statement is called the **semantic deduction theorem**.

Proving logical consequence (2)

- Formula $(F_1 \wedge \cdots \wedge F_n) \rightarrow G$ must be valid, thus its negation $\neg\big((F_1 \wedge \cdots \wedge F_n) \rightarrow G\big) \equiv F_1 \wedge \cdots \wedge F_n \wedge \neg G$ must be inconsistent

Refutation method

Formula G is a logical consequence of $F_1, F_2, \ldots, F_n$ if and only if

$$F_1 \wedge \cdots \wedge F_n \wedge \neg G$$

if **inconsistent** (contradiction). In other words:

$$F_1 \wedge \cdots \wedge F_n \ \vDash \ G$$

if and only if

$$\vDash \neg\big(F_1 \wedge \cdots \wedge F_n \wedge \neg G\big)$$

Proving logical consequence – an example

- Let's prove: $P \lor Q, \neg P \ \vDash \ Q$
- Direct proof:

P	Q	$P \lor Q$	$\neg P$	$\overbrace{(P \lor Q) \land \neg P}^{F}$	$F \to Q$
$\bot$	$\bot$	$\bot$	$\top$	$\bot$	$\top$
$\bot$	$\top$	$\top$	$\top$	$\top$	$\top$
$\top$	$\bot$	$\top$	$\bot$	$\bot$	$\top$
$\top$	$\top$	$\top$	$\bot$	$\bot$	$\top$

- Proof by refutation:

P	Q	$P \lor Q$	$\neg P$	$\overbrace{(P \lor Q) \land \neg P}^{F}$	$\neg Q$	$F \land \neg Q$
$\bot$	$\bot$	$\bot$	$\top$	$\bot$	$\top$	$\bot$
$\bot$	$\top$	$\top$	$\top$	$\top$	$\bot$	$\bot$
$\top$	$\bot$	$\top$	$\bot$	$\bot$	$\top$	$\bot$
$\top$	$\top$	$\top$	$\bot$	$\bot$	$\bot$	$\bot$

Outline

Motivation: Limits of PL

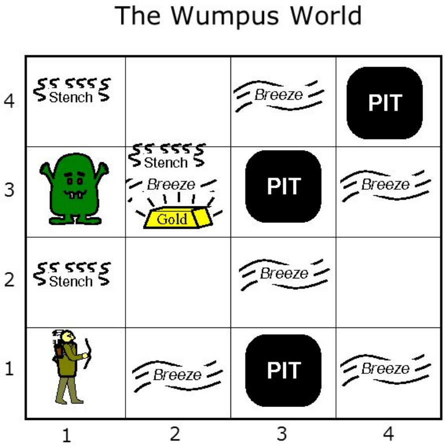
(1) *Every student attends lectures.*

(2) *John is a student.*

(3) *John attends the lectures.*

- We cannot formalize this simple inference in PL
- Why? Because propositions have no internal structure. Here we would have P, Q, R, with no relation between them.
- We need to be able to talk about **relations between objects**: John is a member of students

Example: The Wumpus World



$\neg B(1,1)$
$\neg B(1,2)$
$B(2,1)$
$\neg S(1,1)$
$S(1,2)$
$\neg S(2,1)$
$\neg P(1,1)$
$\neg W(1,1)$
$\neg G(1,1)$

$$B(x,y) \leftrightarrow \big(P(x-1,y) \vee P(x+1,y) \vee P(x,y-1) \vee P(x,y+1)\big)$$
$$S(x,y) \leftrightarrow \big(W(x-1,y) \vee W(x+1,y) \vee W(x,y-1) \vee W(x,y+1)\big)$$

Boosting expressivity

- We need a more expressive system:
  - PL: all there exists is facts that are true or false
  - FOL: there exist **objects** and **relations between objects**
- Relations can be n-ary
  - 0-arity relations: propositions (as we have them in PL)
  - unary relations are for properties (e.g., $STUDENT(John)$)
  - binary relations are for relations between two objects (e.g., $LOVES(John, Marry)$)
  - ternary relations: e.g. $GIVES(John, Marry, flowers)$.
  - ...
- Obviously, FOL subsumes PL
- Unfortunately, increased expressivity comes at a price of **limited provability**

Symbols of FOL

(i) Constants: strings, digits, or lowercased letters from the beginning of the alphabet (e.g., $John$, 123, a, b)

(ii) Variables: lowercased letters from the end of the alphabet (e.g., 'x', 'y', 'z', $\ldots$)

(iii) Functional symbols: lowercased letters (e.g., f, g, h) or lowercased strings (e.g., '$plus$') followed by parentheses

(iv) Predicate symbols: uppercased letters or strings ('A', 'B', 'C',$\ldots$; 'P', 'Q', 'R', '$MOTHER$')

Terms and atoms

Term

(i) A constant is a term
(ii) A variable is a term
(iii) If f is an n-arity functional symbol and $t_1, \ldots, t_n$ are terms, then $f(t_1, \ldots, t_n)$ is a term
(iv) Nothing else is a term

- Examples of terms: 2, 3, $add(3,4)$, $add(x, add(1,4))$

Atom

If P is an n-arity predicate symbol and $t_1, \ldots, t_n$ are terms, then $P(t_1, \ldots, t_n)$ is an **atom**. Nothing else is an atom.

- Examples of atoms: $LOVES(John, Marry)$, $GT(add(1,2), 4)$

Quantifiers

- We use two additional symbols in FOL:
  - $\forall$: universal quantifier (pronounced: "for all")
  - $\exists$: existential quantifier (pronounced: "exists")
- Quantifier scope: the formula to which the quantifier refers (the shortest subformula to the right of the quantifier)
  - E.g., scope of $\forall x$ in $\forall x \exists y P(x, y)$
- **Bound variable:** any variable that occurs within a scope of a quantifier that refers to that very same variable
- **Free variable:** any variable that occurs not bound somewhere in the formula

Formulas

Well-formed formula (wff)

A **well-formed formula** (wff) of FOL is defined recursively as follows:

(1) An atom is a formula

(2) If F is a formula, then $(\neg F)$ is also a formula

(3) If F and G are formulae, then so are $(F \wedge G)$, $(F \vee G)$, $(F \rightarrow G)$, and $(F \leftrightarrow G)$

(4) If F is a formula that <u>contains</u> a variable x that is <u>not bound</u>, then $(\forall x)F$ and $(\exists x)F$ are also formulae

(5) Nothing else is a formula

- Simplifying conventions: Parentheses in (2) may be dropped, those in (3) may be dropped if they are the outer parentheses. Parentheses in (4) may always be dropped.

Formulas

Wff or not?

1. $\forall x(L(x,z))$
2. $(\forall y)(\exists x)P(x,y,z)$
3. $\exists x P(x) \rightarrow Q(a)$
4. $\exists x\Big(P(x) \rightarrow \forall x \forall y Q(x,y)\Big)$
5. $\forall x \forall y\Big(\forall z P(z,x) \rightarrow Q(y,z)\Big)$

Interpretation

Interpretation of a FOL wff

An **interpretation** of a wff consists of:

- a non-empty **domain** D (the things being described)
- a mapping from each constant to an element from D
- a mapping from each n-ary function symbol to a function $D^n \to D$
- a mapping from each n-ary predicate symbol to a function $D^n \to \{\top, \bot\}$

Regarding the last item: the subset of D^n for which a predicate is true is called an **extension** of the predicate

Evaluation

Evaluating the truth value of a formula

Given an interpretation, we can evaluate the truth value of a formula, as follows:

1. For an atom $F(x_1, \dots, x_n)$, the truth value is given by applying the mappings for constants, functions, and predicate symbols
2. Truth values of $(\neg F)$, $(F \wedge G)$, $(F \vee G)$, $(F \rightarrow G)$, $(F \leftrightarrow G)$ are evaluated using truth tables (as in PL)
3. $(\forall x)F$ is evaluated as true iff F is true for every element $d \in D$
4. $(\exists x)F$ is evaluated as true iff F is true for at least one element $d \in D$

- **NB:** We cannot evaluate:
  - formulas that are not wffs!
  - formulas that contain free variables!

Practice time: FOL formula evaluation

Question 1

Evaluate the truth value of the formule $\forall x \exists y P(x, y)$ in an interpretation with the domain $D = \{1, 2\}$ and the following mapping for the predicate P:

$P(1,1)$	$P(1,2)$	$P(2,1)$	$P(2,2)$
$\top$	$\bot$	$\bot$	$\top$

Practice time: FOL formula evaluation

Question 2

We are given an intepretation with the domain $D = \{a, b\}$ and the extension for P defined as $\{(a, a), (b, b)\}$. Evaluate the truth value of the following wffs:

1. $\exists x \forall y P(x, y)$
2. $\exists y \neg P(a, y)$
3. $\forall x \forall y P(x, y)$
4. $\exists x \neg P(x, y)$
5. $\forall x \forall y \Big(P(x, y) \rightarrow P(y, x) \Big)$

FOL equivalences

$$
\begin{array}{rlcl}
[1] & \forall x F[x] & \equiv & \forall y F[y] \\
[2] & \exists x F[x] & \equiv & \exists y F[y] \\
[3] & \neg \forall x F[x] & \equiv & \exists x \neg F[x] \\
[4] & \neg \exists x F[x] & \equiv & \forall x \neg F[x] \\
[5] & \forall x F[x] \vee \forall x G[x] & \equiv & \forall x F[x] \vee \forall y G[y] \\
[6] & \forall x F[x] \vee \exists x G[x] & \equiv & \forall x F[x] \vee \exists y G[y] \\
[7] & \exists x F[x] \vee \forall x G[x] & \equiv & \exists x F[x] \vee \forall y G[y] \\
[8] & \exists x F[x] \vee \exists x G[x] & \equiv & \exists x F[x] \vee \exists y G[y] \\
[9] & \forall x F[x] \wedge \forall x G[x] & \equiv & \forall x F[x] \wedge \forall y G[y] \\
[10] & \forall x F[x] \wedge \exists x G[x] & \equiv & \forall x F[x] \wedge \exists y G[y] \\
[11] & \exists x F[x] \wedge \forall x G[x] & \equiv & \exists x F[x] \wedge \forall y G[y] \\
[12] & \exists x F[x] \wedge \exists x G[x] & \equiv & \exists x F[x] \wedge \exists y G[y] \\
[13] & \forall x F[x] \vee \forall y G[y] & \equiv & \forall x \forall y (F[x] \vee G[y]) \\
[14] & \forall x F[x] \wedge \forall y G[y] & \equiv & \forall x \forall y (F[x] \wedge G[y])
\end{array}
$$

$F[x]$ and $G[x]$ denotes formulas containing x

FOL equivalences

$$
\begin{array}{lrcl}
[15] & \forall x F[x] \lor H\{x\} & \equiv & \forall x (F[x] \lor H\{x\}) \\
[16] & \forall x F[x] \land H\{x\} & \equiv & \forall x (F[x] \land H\{x\}) \\
[17] & \exists x F[x] \lor H\{x\} & \equiv & \exists x (F[x] \lor H\{x\}) \\
[18] & \exists x F[x] \land H\{x\} & \equiv & \exists x (F[x] \land H\{x\}) \\
[19] & \forall x (F[x] \land G[x]) & \equiv & \forall x F[x] \land \forall x G[x] \\
[20] & \forall x (F[x] \land G[x]) & \equiv & \forall x F[x] \land \forall y G[y] \\
[21] & \forall x (F[x] \land G[x]) & \equiv & \forall x \forall y (F[x] \land G[y]) \\
[22] & \exists x (F[x] \lor G[x]) & \equiv & \exists x F[x] \lor \exists x G[x] \\
[23] & \exists x (F[x] \lor G[x]) & \equiv & \exists x F[x] \lor \exists y G[y] \\
[24] & \exists x (F[x] \lor G[x]) & \equiv & \exists x \exists y (F[x] \lor G[y])
\end{array}
$$

$H\{x\}$ denotes a formula <u>not</u> containing x

FOL undecidability

- FOL **validity**, **inconsistency**, and **satisfiability** are defined in the same manner as in PL
- For a PL formula with n variables, we have 2^n interpretations (why?)
- **Q**: How many interpretations does a FOL formula have?
- **A**: Infinitely many!
- There is infinitely many possible domains D (and some of them are infinite, too!)
- Therefore, there are no truth tables in FOL!
- **Q:** How does this affect the semantics? (Think: logical consequence)
- **A:** We can't prove logical consequences by exhaustive enumeration of all interpretations $\Rightarrow$ **undecidability**
- Because enumeration won't work, so we have to look for more efficient procedures $\Rightarrow$ **proof theory**
- Regardless what method we use, it will be limited by **undecidability**

Outline

Mapping natural language to FOL

1. *John is a hardworking student.*
2. *All students are clever.*
3. *No student is clever.*
4. *Some students are clever.*

Mapping natural language to FOL

1. *John is a hardworking (H) student (S).*
   - $H(John) \wedge S(John)$ ✓
   - $HS(John)$
   - HSJ
   - $\forall x \big(John(x) \rightarrow (H(x) \wedge S(x)) \big)$

Mapping natural language to FOL

2. *All students are clever (C).*

   - $\forall x\big(S(x) \wedge C(x)\big)$ ✗
   - $\forall student\ C(student)$ ✗
   - $\forall x \in student\ C(x)$ ✗
   - $\forall x\big(S(x) \rightarrow C(x)\big)$ ✓
   - $\forall x S(x) \rightarrow C(x)$ ✗

Mapping natural language to FOL

3. *No student is clever.*

- $\forall x \big(S(x) \rightarrow \neg C(x)\big)$ ✓
- $\forall x \neg \big(S(x) \land C(x)\big)$ ✓
- $\neg \exists x \big(S(x) \land C(x)\big)$ ✓
- $\neg \forall x \big(S(x) \rightarrow C(x)\big)$ ✗
- $\neg \forall x \big(S(x) \land C(x)\big)$ ✗
- $\forall x \big(\neg S(x) \land \neg C(x)\big)$ ✗

Mapping natural language to FOL

4. *Some students are clever.*

   - $\exists x\big(S(x) \land C(x)\big)$ ✓
   - $\exists x\big(S(x) \rightarrow C(x)\big)$ ✗

Mapping natural language to FOL

1. *John is a hardworking (H) student (S).*

$$H(John) \land S(John)$$

2. *All students are clever (C).*

$$\forall x\big(S(x) \to C(x)\big)$$

3. *No student is clever.*

$$\forall x\big(S(x) \to \neg C(x)\big) \;\equiv\; \forall x \neg\big(S(x) \land C(x)\big) \;\equiv\; \neg\exists x\big(S(x) \land C(x)\big)$$

4. *Some students are clever.*

$$\exists x\big(S(x) \land C(x)\big)$$

More details in (HR only): `https://goo.gl/15neLq`

Example: Customs officers and diplomats

Premises

Customs officers searched everyone who entered the country but wasn't a diplomat. Some smugglers who entered the country were searched only by smugglers. No smuggler is a diplomat.

Conclusion

Some customs officers are smugglers.

Mapping natural language to FOL

Customs officers (O) searched (S) everyone who entered (E) the country but wasn't a diplomat (D).

$$\forall x\Big(\big(E(x) \land \neg D(x)\big) \rightarrow \exists y\big(O(y) \land S(y,x)\big)\Big)$$

Some smugglers (M) who entered the country were searched only by smugglers.

$$\exists x\Big(M(x) \land E(x) \land \forall y\big(S(y,x) \rightarrow M(y)\big)\Big)$$

No smuggler is a diplomat.

$$\forall x\big(M(x) \rightarrow \neg D(x)\big)$$

Some customs officers are smugglers.

$$\exists x\big(O(x) \land M(x)\big)$$

Some limitations of FOL

- Modelling of **time** and **events**
  - Alternatives: situational calculus, event calculus
- Modelling of **mental states**
  - propositional attitudes: beliefs, desires, intentions

$$Believes\big(Lois, CanFly(Superman)\big)$$

  - Problem: mental states are not **referentially transparent** $\Rightarrow$ substitution of equivalent expressions can alter the formula's meaning!

$$(Superman = Clark) \land Believes\big(Lois, CanFly(Superman)\big)$$
$$\vDash CanFly\big(Lois, CanFly(Clark)\big)$$

  - Solution: **modal logic** (more complex semantic model based on **possible worlds**)

Wrap-up

- There are many kinds of logic that differ in **expressivity**
- Every logical system consists of **(1) syntax, (2) semantics, and (3) a proof theory**
- A formula can be **consistent**, **valid**, or **inconsistent**
- **Logical consequence** is true whenever the premises are true. We can prove this it either **directly** or by **refutation**
- In propositional logic we can express propositions that are either true or false, and formulas have finite number of interpretations
- In PL, the logical consequence can be proven using a **truth table**
- In first-order logic (FOL), we can express properties and relations, however the number of interpretations is infinite

Next topic: Automated reasoning