ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

A common language: First Order Logic

Prof. Ing. Federico Chesani

DISI

Department of Informatics – Science and Engineering

Motivation

- Usually, we communicate each other using Natural Language
- E.g.: these slides are in NL (in English, to be precise)

- Natural language is subject, by its own nature, to **ambiguity**

- We might not like ambiguity
  - E.g., students would not *perfectly* understand what I am teaching them… they would have a higher risk of failing the exam.
  - In this course we will discuss about knowledge and reasoning: the risk of ambiguities is higher

Goal of this lesson

- We need to agree to a language that is free of ambiguity
- Choice: **First Order Logic**, or better saying, logic in general
  - Bertrand Russel, when approaching the writing of Principia Mathematica, came up with the same conclusion…

MIND!!! We will use a logic formalism in two different manners:

1. as a tool for explaining problems, issues, and solutions related to knowledge and reasoning
2. as a tool per-sé for knowledge representation and reasoning

Disclaimer and Further reading

- These slides are heavily inspired by slides of Prof. Gabbrielli

Further reading:

- Slides and course by Prof. Gabbrielli
- AIMA
  - chapter 7 for propositional logic and resolution
  - chapter 8 for intro to First Order Logic
  - chapter 9 for First Order Inference

A notation for representing reasoning: symbols, the act of reasoning, interpretations

A notation for representing the act of reasoning

We have a "bootstrap" problem:

- we want to discuss on how to represent knowledge and reasoning
- better, we want to discuss "systems" for representing knowledge and do some (automatic) reasoning (and determine their properties, how they are made, etc.)

… but we are missing any way for representing the object of our dialogues…

A notation for representing the act of reasoning

Solution:

- Let us use **symbols**
  - Symbols can be anything, for us they will be signs drawn on a piece of paper or a slide
  - We will associate a symbol to a concept (more on this, later)
  - We will use symbols together and create *sentences*
  - *Sentences, sometime, might be true or false*

- We will represent the "**act of reasoning**" in the following way:
  - we will write the things that we know (or believe we know, or that we know they are true)
  - a horizontal line
  - below the line, the result of our reasoning act

A notation for representing the act of reasoning

"It's raining cats and dogs"

"I'll get wet"

"I am hungry"

"I'll have a sandwich"

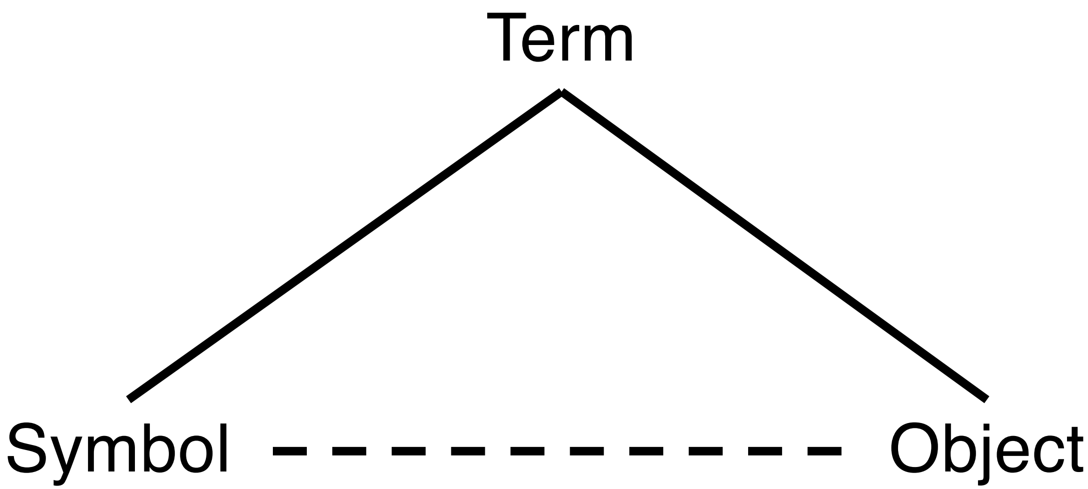
"Every human is mortal" "Socrates is human"

"Socrates is mortal"

The crossing of the line from above to below is the act of reasoning

And the symbols?

- Historically, a long discussed topic. We will refer to the semiotic triangle by Ogden and Richards:



Example:

- a symbol "pizza" made of five graphical traits
- my personal idea of pizza
- the pizza I will have tonight, tangible, physical, eatable, gorgeous, luxurious pizza

The interpretation of symbols

An immediate problem: who decides the links between symbols and ideas/objects?

- Everyone in this classroom has the right to decide the meaning of a symbol
- Everyone in this classroom has its own, rightful *interpretation* of symbols
- The *truth* of a sentence depends on the meaning of the symbols

- If anyone can have her/his own interpretation, what happens to the "correctness" of the reasoning?

"It's raining cats and dogs"

"I'll get wet"

"It's raining cats and dogs"

"Let's save the kitten!!!"

The interpretation of symbols

An interpretation $\mathfrak{I}$ is a pair (D, I) where:

- D is any set of objects, called domain
- I is a mapping called the interpretation mapping, from the (non logical) symbols to elements and relations over D

Given a set of sentences, their truth value will change depending on the interpretation we will give to the symbols: there is a strict relation between the notion of interpretation and the truth value.

Can we reason, then?

The interpretation of symbols

Example:

"Brave men run in my family."

Bob Hope as "Painless" Peter Potter in *The Paleface*, 1948

Which is interpretation of "run"?

$\mathfrak{I}_1$: run as the act of "walking fast"

The sentence is false: I am certainly brave, but I usually don't run... ;-)

$\mathfrak{I}_2$: run as "belong", "being part of"

The sentence might be true or false.

The idea of Logical Consequence

To avoid the problem of the many different interpretations of symbols, we will restrict:

- from the notion of truth (something more related to philosophy)…
- … to the notion of Logical Consequence

Let $\Gamma = F_1, \dots, F_n$ be a set of sentences (hypotheses or premises, and F be a sentence (conclusion).

F is a **logical consequence** of Γ, written $\Gamma \vDash F$, if for ***any possible different interpretation***, it is always true that if all the formulas in Γ are true (in the world under examination) then also F is true.

Propositional Logic

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Propositional Logic

Propositional logic is the simplest logic – it illustrates basic ideas using propositions Pi

- Each Pi is referred as an atom or atomic formula
- Atoms Pi can be connected together by connectives
  - $\wedge$, $\vee$, $\rightarrow$, $\neg$, $\leftrightarrow$, $\perp$
- A well-formed formula is:
  - an atomic formula Pi
  - if A and B are well-formed formulas, then $A \wedge B$, $A \vee B$, $A \rightarrow B$, and $\neg A$ are well-formed formulas

Propositional Logic: Semantics

Each atom Pi can be either **true** or **false**

("tertium non datur": why only two values of truth?)

Given a propositional formula G, let $\{A_1, \dots, A_n\}$ be the set of atoms which occur in the formula, an ***interpretation*** $\mathfrak{I}$ of G is an assignment of truth values to $\{A_1, \dots, A_n\}$.

What about the non-atomic formula G? Interpretation is extended through the "truth tables" for the logical connectives.

Propositional Logic: Semantics

- Given an interpretation $\mathfrak{I}$, a formula G is said to be true in $\mathfrak{I}$ iff G is evaluated to true in the interpretation
- Given a formula G and an interpretation $\mathfrak{I}$, if G is true in $\mathfrak{I}$ we say that $\mathfrak{I}$ is a ***model*** for G, and usually write

$$\mathfrak{I} \vDash G$$

There can be "extreme situations":

- no matter how we will choose an interpretation, G is always true: G is a valid formula, or a tautology ($\vDash G$)
- no matter how we will choose an interpretation, G is always false: G is an inconsistent formula
- mind: *invalid* is different from *inconsistent*.

More frequently, there will be formulas G that are invalid, bur are satisfiable (i.e., consistent).

Propositional Logic and Decidability

- Propositional Logic is decidable: there is always a terminating method to decide whether a formula is valid.

How to decide if a formula is valid or not?

1. enumerate all the possible interpretations
2. look at the formula for each interpretation

Which (computational) cost?

Example: a small problem of logic network for controlling a railway station with one train track, can be represented with 30,000 propositional atoms. Determining properties about such a network would consist to explore up to $2^{30,000}$ different interpretations.

Propositional Logic and Decidability

Propositional Logic is decidable…

… but this does not help us to efficiently decide.

WAIT! which is our goal?

- Goal: to decide if G is a valid formula

… nice, but stated in this way, not really interesting.

CHANGE GOAL:

- Goal: to decide if, given some formulas are true, a formula G is true as well
  - it is a reformulation of the goal above: we could always substitute each true atom with true, simplify G, and check if G is valid for all the possible interpretations.
  - this perspective allows us to introduce the notion of logical consequence

Logical consequence, Logical equivalence

- Given a set of formulas $\{F_1, \ldots, F_n\}$ and a formula G, G is said to be a **logical consequence** of $F_1, \ldots, F_n$ iff for *any interpretation* $\mathfrak{I}$ in which $F_1 \wedge \cdots \wedge F_n$ is true, G is also true. Syntactically: $F_1 \wedge \cdots \wedge F_n \vDash G$

- Def.: Two formulas F and G are **logically equivalent** **F ≡ G** iff the truth values of F and G are the same under *every interpretation* of F and G

- $\mathrm{F} \equiv Q$ iff $F \vDash Q$ and $Q \vDash F$

Logical equivalence

$$
\begin{array}{rcll}
(\alpha \land \beta) & \equiv & (\beta \land \alpha) & \text{commutativity of } \land \\
(\alpha \lor \beta) & \equiv & (\beta \lor \alpha) & \text{commutativity of } \lor \\
((\alpha \land \beta) \land \gamma) & \equiv & (\alpha \land (\beta \land \gamma)) & \text{associativity of } \land \\
((\alpha \lor \beta) \lor \gamma) & \equiv & (\alpha \lor (\beta \lor \gamma)) & \text{associativity of } \lor \\
\neg(\neg\alpha) & \equiv & \alpha & \text{double-negation elimination} \\
(\alpha \Rightarrow \beta) & \equiv & (\neg\beta \Rightarrow \neg\alpha) & \text{contraposition} \\
(\alpha \Rightarrow \beta) & \equiv & (\neg\alpha \lor \beta) & \text{implication elimination} \\
(\alpha \Leftrightarrow \beta) & \equiv & ((\alpha \Rightarrow \beta) \land (\beta \Rightarrow \alpha)) & \text{biconditional elimination} \\
\neg(\alpha \land \beta) & \equiv & (\neg\alpha \lor \neg\beta) & \text{De Morgan} \\
\neg(\alpha \lor \beta) & \equiv & (\neg\alpha \land \neg\beta) & \text{De Morgan} \\
(\alpha \land (\beta \lor \gamma)) & \equiv & ((\alpha \land \beta) \lor (\alpha \land \gamma)) & \text{distributivity of } \land \text{ over } \lor \\
(\alpha \lor (\beta \land \gamma)) & \equiv & ((\alpha \lor \beta) \land (\alpha \lor \gamma)) & \text{distributivity of } \lor \text{ over } \land
\end{array}
$$

Figure 7.11 Standard logical equivalences. The symbols α, β, and γ stand for arbitrary sentences of propositional logic.

From Russel Norvig, AIMA, 4th edition.

Reasoning

- The notion of logical consequence allows us to "forget" the problem of the many different interpretations

- Is there a way that allows us to deal with the formulas, and to decide if a specific formula G is valid?

- Many different methods for working with the symbols have been identified. Working with the symbols is indeed indicated as reasoning.
  - usually, we refer to a reasoning engine/method, and use the syntax $F_1 \wedge \cdots \wedge F_n \vdash^E G$ to denote that G can be deduced by $F_1 \wedge \cdots \wedge F_n$ using the reasoning method E
  - in general, we will write $KB \vdash^E G$ where KB is a knowledge base in some form/language/notation/logic

Reasoning

1. **Which reasoning engine/method?**
   - "Natural Deduction" $F_1 \wedge \cdots \wedge F_n \vdash^{ND} G$
   - "Refutation and Resolution" $F_1 \wedge \cdots \wedge F_n \vdash^{RR} G$
   - ...
2. **Does a method always bring "correct" answers?**
   - there are methods that are very useful and interesting, but that might provide wrong answers.
   - a method E is said to be **Sound** if it holds: $(\Gamma \vdash^{E} \varphi) \Rightarrow (\Gamma \vDash \varphi)$
   - e.g., abduction and induction are a very common way of human reasoning, but they might not be sound
3. **Does a method allow to prove all the possible φ ?**
   - a method E is said to be **Complete** if it holds:

$$(\Gamma \vDash \varphi) \Rightarrow (\Gamma \vdash^{E} \varphi)$$

Reasoning and soundness

The Inductivist Turkey

This example is attributed to B. Russel, 1912; it was originally a chicken; it became a turkey only later, when other philosophers re-use it.

*The turkey found that, on his first morning at the turkey farm, he was fed at 9 a.m. Being a good inductivist turkey he did not jump to conclusions. He waited until he collected a large number of observations that he was fed at 9 a.m. and made these observations under a wide range of circumstances, on Wednesdays, on Thursdays, on cold days, on warm days. Each day he added another observation statement to his list. Finally, he was satisfied that he had collected a number of observation statements to inductively infer that "**I am always fed at 9 a.m.**".*

However, on the morning of Christmas eve…

- Which are the sentences known to be true?
- Which is the conclusion?

Reasoning by deduction

Theorem: Given a set of formulas $\{F_1, \dots, F_n\}$ and a formula G,

$F_1 \wedge \cdots \wedge F_n \vDash G$ if and only if $\vDash (F_1 \wedge \cdots \wedge F_n) \rightarrow G$

How to use it for reasoning?

We can prove the logical consequence of G by proving the validity of $(F_1 \wedge \cdots \wedge F_n) \rightarrow G$.

Reasoning by refutation

Theorem: Given a set of formulas $\{F_1, \ldots, F_n\}$ and a formula G,

$F_1 \wedge \cdots \wedge F_n \vDash G$ if and only if $F_1 \wedge \cdots \wedge F_n \wedge \neg G$ is inconsistent.

How to use it for reasoning?

We can prove the logical consequence of G by proving the inconsistency of $F_1 \wedge \cdots \wedge F_n \wedge \neg G$.

Reasoning by refutation – how to?

The idea of using refutation for reasoning was presented in:

Robinson , J. Alan (1965): **A Machine-Oriented Logic Based on the Resolution Principle**, Journal of ACM, 12 (1): 23–41.

To do so, the author proposed to exploit the Resolution rule, together with the refutation mechanism.

Resolution

- Resolution rule applies to logic formulas expressed in the form of clauses
- A clause is a disjunction of literals, i.e. atoms and negated atoms: $A_1 \vee \cdots \vee A_n \vee \neg B_1 \vee \cdots \vee \neg B_m$
- Let us suppose to have two clauses such that:
  - $C_1 \equiv A_1 \vee \cdots \vee A_{i-1} \vee A_i \vee A_{i+1} \vee \cdots \vee A_n$
  - $C_2 \equiv B_1 \vee \cdots \vee B_{j-1} \vee \neg A_i \vee B_{j+1} \vee \cdots \vee B_m$
- Then the following clause

$$C_3 \equiv A_1 \vee \cdots \vee A_{i-1} \vee A_{i+1} \vee \cdots \vee A_n \vee B_1 \vee \cdots \vee B_{j-1} \vee B_{j+1} \vee \cdots \vee B_m$$

is a **logical consequence** of C1 and C2.

Resolution

$$\text{Resolution} \quad \frac{(\boldsymbol{A} \vee \boldsymbol{B}) \qquad (\neg \boldsymbol{B} \vee \boldsymbol{C})}{(\boldsymbol{A} \vee \boldsymbol{C})}$$

A	B	C	$(A \vee B)$	$(\neg B \vee C)$	$(A \vee C)$
F	F	F	F	T	F
F	F	T	F	T	T
F	T	F	T	F	F
F	**T**	**T**	**T**	**T**	**T**
T	**F**	**F**	**T**	**T**	**T**
T	**F**	**T**	**T**	**T**	**T**
T	T	F	T	F	T
T	**T**	**T**	**T**	**T**	**T**

Remember we are interested in **logical consequence!**

From few slides ago:

Given a set of formulas $\{F_1, \ldots, F_n\}$ and a formula G, G is said to be a **logical consequence** of $F_1, \ldots, F_n$ iff for *any interpretation* $\mathfrak{I}$ in which $F_1 \wedge \cdots \wedge F_n$ is true, G is also true. Syntactically: $F_1 \wedge \cdots \wedge F_n \vDash G$

Refutation and Resolution together – summary

- Goal: we want to know if $F_1 \wedge \cdots \wedge F_n \vDash G$
- From theorem, we know that we can prove the above iff we prove $F_1 \wedge \cdots \wedge F_n \wedge \neg G$ is inconsistent

A possible strategy:

- Change our goal: prove that $F_1 \wedge \cdots \wedge F_n \wedge \neg G$ is inconsistent, i.e. there is no possible interpretation $\mathfrak{I}$ that makes true the formula.
- How? through resolution, we derive the inconsistency

Refutation and Resolution together – practical steps

1. Transform our knowledge base KB in CNF form: each conjunct will be a clause
2. Negate the formula G and add it to the KB
3. Apply the resolution principle (to all the clauses) until we reach the inconsistency

function PL-RESOLUTION(KB, α) **returns** *true* or *false*
  inputs: KB, the knowledge base, a sentence in propositional logic
      α, the query, a sentence in propositional logic

  $clauses \leftarrow$ the set of clauses in the CNF representation of $KB \wedge \neg\alpha$
  $new \leftarrow \{\}$
  while *true* **do**
    for each pair of clauses C_i, C_j **in** *clauses* **do**
      $resolvents \leftarrow$ PL-RESOLVE(C_i, C_j)
      if *resolvents* contains the empty clause **then return** *true*
      $new \leftarrow new \cup resolvents$
    if $new \subseteq clauses$ **then return** *false*
    $clauses \leftarrow clauses \cup new$

Figure 7.13 A simple resolution algorithm for propositional logic. PL-RESOLVE returns the set of all possible clauses obtained by resolving its two inputs.

From Russel Norvig, AIMA, 4th edition.

Refutation and Resolution together – example

Exercise:

Given the following

$$KB = \{(a \rightarrow c \vee d) \wedge (a \vee d \vee e) \wedge (a \rightarrow \neg c)\}$$

Prove that $G = \{(d \vee e)\}$ is logical consequence of KB.

Solution:

i. Transform in CNF the KB_{CNF}:

1. $(\neg a \vee c \vee d)$
2. $(a \vee d \vee e)$
3. $(\neg a \vee \neg c)$

ii. Negate the formula G:

4. $\neg d$
5. $\neg e$

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Refutation and Resolution together – example

iii. Derive the possible resolvents:

First iteration:

6. $(1+2)$: $(c \vee d \vee e)$
7. $(2+3)$: $(\neg c \vee d \vee e)$
8. $(1+4)$: $(\neg a \vee c)$
9. $(2+4)$: $(a \vee e)$
10. $(2+5)$: $(a \vee d)$
11. $(1+3)$: $(\neg a \vee d)$

Second iteration:

12. $(10+11)$: d

Third iteration:

13. $(4+12)$: empty clause -> contradiction -> the original formula G is logical consequence of the KB.

Refutation and Resolution – properties for the propositional setting

- Sound? if $KB \cup \{\neg G\} \vdash^{RR} \perp$ then $\mathrm{KB} \vDash G$
- Complete? if $KB \vDash G$ then $KB \cup \{\neg G\} \vdash^{RR} \perp$
- Decidable.
- Efficient? not really, np-Complete in the worst case.
- Expressive power? poor, unsatisfactory…

- Things are better if we restrict to Horn Clauses (at most one positive literal)…
- … and we adopt the SLD rule (Linear resolution with Selection function for Definite clauses).

First Order Logic

First Order Logic – FOL

Five different sets of elements:

- The set of the constants (e.g., "federico")
- The set of function symbols (e.g., son_of(federico))
- The set of variables (e.g., X)
- The set of predicate symbols
- The set of logical connectives
  - ¬ (negation),
  - ∧ (conjunction),
  - ∨ (disjunction),
  - → (implication),
  - ↔(equivalence)
  - brackets "(" ")"
  - Existential ∃ and universal ∀ quantificators.

First Order Logic – Terms

A term is recursively defined as:

- A constant;
- A variable;
- If f is a n-ary function symbol, and $t_1, \dots, t_n$ are terms, then $f(t_1, \dots, t_n)$ is also a term.

Terms are introduced and used as symbols to indicate the objects of our discourse

E.g.:

- federico
- francesco
- son_of(federico) === francesco

First Order Logic – Atoms or Atomic formulas

An atom/atomic formula is defined as the application of a n-ary predicate symbol p to n terms $p(t_1, \ldots, t_n)$.

Predicates are introduced and used to reason about the truth of sentences

E.g.:

- awesome(federico)
- teenager(francesco)
- loves(federico, francesco)

First Order Logic – Well Formed Formulas

Well Formed Formulas (WFF) are recursively defined as:

- Every atom is a WFF
- If φ and ψ are WFF, then the following are WFF:
  - $\neg\varphi$
  - $\varphi \wedge \psi$
  - $\varphi \vee \psi$
  - $\varphi \rightarrow \psi$
  - $\varphi \leftrightarrow \psi$
- if φ is a WFF and X is a variable, then the following are WFF:
  - $\exists X\ \varphi$
  - $\forall X\ \varphi$

- A Literal is defined to be an atom or its negation.

FOL – How to reason?

Reasoning in FOL can be done using several ways

- Refutation and resolution (Robinson 1965) is adopted, with some changes w.r.t. the mechanism seen for PL (see the slides by Prof. Gabbrielli)
  - variables, quantifiers and Skolemization, unification and mgu, etc.

- FOL Logical Consequence is semi-decidable (Church, Turing 1936, Godel 1930)
  - if a formula is a theorem of a set of axioms, then there exists a mechanism that will derive the theorem in a finite number of steps;
  - if it is not, there is no mechanism that guarantee always the termination

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

federico.chesani@unibo.it