Logic: A Summary

Jacek Malec
Dept. of Computer Science, Lund University, Sweden

February 8, 2017

Formal languages and syntax:

propositional variables: P, Q, R, S
operators (connectives): $\neg$, $\vee$, $\wedge$
formulae: P, $\neg Q \wedge R$, $\neg(Q \vee R)$
Language:
the set of all well-formed formulae (wff):

$$\{P, Q, \neg P, \neg Q, P \wedge Q, P \vee Q, \ldots\}$$

Assigning truth values to symbols:

P is TRUE
Q is FALSE

Interpretation: an assignment to *all* of the variables.
It determines the truth values for more complex formulae:

$$\neg P \vee Q$$

$\neg P \vee P$
a tautology
$\neg P \wedge P$
a contradiction

Logical equivalence:

$$Q \vee \neg P$$
$$\neg Q \vee P$$

$$\neg P \vee P$$
$$\neg P \wedge P$$

$$P \vee Q$$
$$\neg(\neg P \wedge \neg Q)$$

$$\neg P \vee Q$$
$$P \rightarrow Q$$

Formal systems:

- Axioms
- Axiom schemas
- Rules of inference

Rules of inference:

Modus Ponens:

$$\frac{\begin{array}{c} A \\ A \rightarrow B \end{array}}{B}$$

Conjunction:

$$\frac{\begin{array}{c} A \\ B \end{array}}{A \wedge B}$$

Theoremhood:

1. $P \rightarrow Q$
   assume this is given as true
2. $Q \rightarrow R$
   assume this is given as true
3. P
   assume this is given as true
4. Q
   Modus Ponens using 1 and 3
5. R
   Modus Ponens using 2 and 4

Lines 1–4 constitute a *proof* of Q.
Lines 1–5 constitute a proof of R.
Q is a *theorem*.

Satisfiability:

Is there an assignment to the variables such that the following formula is true?

$$\neg P \wedge (Q \vee \neg(R \wedge \ldots))$$

Satisfiability problem is $O(2^n)$
Similar questions:

- Is it a tautology?
- Is it a contradiction?

Knowledge representation:

P = ($<$ (temp pump45) 85 degrees Celsius)
Q = (correctly_functioning pump45)

$$P \to Q$$

Knowledge representation:

P = ($<$ (temp pump45) 85 degrees Celsius)
Q = (correctly_functioning pump45)

$$P \to Q$$

$B_{1,1}$ = no breeze in (1, 1)
$P_{1,2}$ = no pit in (1, 2)
$P_{2,1}$ = no pit in (2, 1)

$$B_{1,1} \to P_{1,2} \vee P_{2,1}$$

Knowledge representation:

P = ($<$ (temp pump45) 85 degrees Celsius)
Q = (correctly_functioning pump45)

$$P \to Q$$

$B_{1,1}$ = no breeze in (1, 1)
$P_{1,2}$ = no pit in (1, 2)
$P_{2,1}$ = no pit in (2, 1)

$$B_{1,1} \to P_{1,2} \vee P_{2,1}$$

$$B_{1,1} \leftarrow P_{1,2} \vee P_{2,1}$$

Knowledge representation:

P = ($<$ (temp pump45) 85 degrees Celsius)
Q = (correctly_functioning pump45)

$$P \to Q$$

$B_{1,1}$ = no breeze in (1, 1)
$P_{1,2}$ = no pit in (1, 2)
$P_{2,1}$ = no pit in (2, 1)

$$B_{1,1} \to P_{1,2} \vee P_{2,1}$$

$$B_{1,1} \leftarrow P_{1,2} \vee P_{2,1}$$

$$B_{1,1} \leftrightarrow P_{1,2} \vee P_{2,1}$$

Expert or Rule-Based Systems:

```
(if (and p1 p2 ... pn) q)
```

Tasks:

- prediction

```
(if (and john_is_in_the_building
         (not john_is_in_his_office)
         (not john_is_in_the_copy_room))
    john_is_in_the_conference_room)
```

- diagnosis

```
(if
 (and engine_is_running_hot
   engine_coolant_levels_within_spec)
 evidence_of_a_lubrication_problem)
```


A note on Resolution:

It is a generalization of Modus Ponens

$$\frac{\begin{array}{c} A_1 \vee A_2 \vee \ldots \vee \neg C \vee \ldots \vee A_m \\ B_1 \vee B_2 \vee \ldots \vee C \vee \ldots B_n \end{array}}{A_1 \vee A_2 \vee \ldots \vee A_m \vee B_1 \vee B_2 \vee \ldots \vee B_n}$$

Modus Ponens:

$$\frac{\begin{array}{l} \neg P \ \ \vee Q \\ \ P \end{array}}{Q}$$

First Order Predicate Logic: Syntax

- **Predicates** (relations, properties):

First Order Predicate Logic: Syntax

- **Predicates** (relations, properties): *AgeOf*, *Bald*, *CapitalOf*, *YoungerThan*, $<$, $=$, P, Q, ...
- **Constants**:

First Order Predicate Logic: Syntax

- **Predicates** (relations, properties):
  AgeOf, *Bald*, *CapitalOf*, *YoungerThan*, $<$, $=$, *P*, *Q*, ...
- **Constants**:
  Jacek, 59, *Stockholm*, *Lund*, *Sweden*, *Pierre*, *table*59, *c*, *d*, ...
- **Functions**:

Logic

First Order Predicate Logic: Syntax

- **Predicates** (relations, properties):
  AgeOf, *Bald*, *CapitalOf*, *YoungerThan*, $<$, $=$, *P*, *Q*, ...
- **Constants**:
  Jacek, 59, *Stockholm*, *Lund*, *Sweden*, *Pierre*, *table*59, *c*, *d*, ...
- **Functions**: *fatherOf*, *ageOf*, *lengthOf*, *locationOf*, ...
- **Terms**: constants, variables, functions thereof
- **Atomic sentences**: relation over appropriate amount of terms
  $AgeOf(Jacek, 59)$, $Bald(Jacek)$, $8 < x$,
  $YoungerThan(Jacek, fatherOf(Jacek))$, $P(x, y, z)$
  $locationOf(TJR048) =$ PDammgården, ...

Logic

First Order Predicate Logic: Syntax

- **Predicates** (relations, properties):
  AgeOf, *Bald*, *CapitalOf*, *YoungerThan*, $<$, $=$, *P*, *Q*, ...
- **Constants**:
  Jacek, 59, *Stockholm*, *Lund*, *Sweden*, *Pierre*, *table*59, *c*, *d*, ...
- **Functions**: *fatherOf*, *ageOf*, *lengthOf*, *locationOf*, ...
- **Terms**: constants, variables, functions thereof
- **Atomic sentences**: relation over appropriate amount of terms
  $AgeOf(Jacek, 59)$, $Bald(Jacek)$, $8 < x$,
  $YoungerThan(Jacek, fatherOf(Jacek))$, $P(x, y, z)$
  $locationOf(TJR048) =$ PDammgården, ...
- **Well-formed formulae**: as before plus
  $\forall xA$ and $\exists xA$ are wffs if A is a wff

Quantifiers:

$$\forall x(swedish - citizen(x) \rightarrow has - pnr(x))$$

$$\exists y(polish - citizen(y) \wedge has - pnr(y))$$

$\forall xA$ and $\exists xA$ are wffs if A is a wff

- scope of a quantifier
- free variable
- closed formula
- ground formula

Formal System for FOPC:

language of FOPC, axioms + RES and UI

where Universal Instantiation:

$$\frac{\forall x A}{A'(x \rightarrow t)}$$

e.g. from

$$\forall x, y(Pit(x, y) \rightarrow Breeze(x, y + 1) \wedge Breeze(x + 1, y))$$

we can infer

$$Pit(1, 2) \rightarrow Breeze(1, 3) \wedge Breeze(2, 2),$$

Logic

Formal System for FOPC:

language of FOPC, axioms + RES and UI

where Universal Instantiation:

$$\frac{\forall x A}{A'(x \rightarrow t)}$$

e.g. from

$$\forall x, y(Pit(x, y) \rightarrow Breeze(x, y + 1) \wedge Breeze(x + 1, y))$$

we can infer

$$Pit(1, 2) \rightarrow Breeze(1, 3) \wedge Breeze(2, 2),$$

and

$$Pit(2, 1) \rightarrow Breeze(2, 2) \wedge Breeze(3, 1),$$

and ...

Theories:

$$\forall x, y \neg clown(x) \vee loves(y, x)$$

Everybody loves a clown.

$$\forall x, y \neg winner(x) \vee \neg game(y) \vee \neg plays(x, y) \vee wins(x, y)$$

A winner wins every game (s)he plays.

Pattern:

$$\forall x_1, ..., x_n A$$

where *A* is in CNF

Logically equivalent formulae:

1.

$$\forall x, y(clown(x) \rightarrow loves(y, x))$$

$$\forall x(clown(x) \rightarrow \forall y(loves(y, x)))$$

2.

$$\forall x A \leftrightarrow \neg \exists x \neg A$$

$$\exists x A \leftrightarrow \neg \forall x \neg A$$

Example:

$$(\forall x, y) \neg clown(x) \vee loves(y, x)$$

$$(\forall y) \neg((\exists x)(clown(x) \wedge \neg loves(y, x)))$$

$$(\forall x) clown(x) \rightarrow \neg((\exists y) \neg loves(y, x))$$

Theorem proving:

Show $loves(Pia, Kalle)$ given axioms:

1. $\forall x, y clown(x) \rightarrow loves(y, x)$
2. $clown(Kalle)$

Proof:

1. $\forall x, y clown(x) \rightarrow loves(y, x)$ (AXIOM)
2. $clown(Kalle)$ (AXIOM)
3. $\forall y clown(Kalle) \rightarrow loves(y, Kalle)$
   UI $x \rightarrow Kalle$
4. $clown(Kalle) \rightarrow loves(Pia, Kalle)$
   UI $y \rightarrow Pia$
5. $loves(Pia, Kalle)$
   MP 2,4

Search, search everywhere...

Theorem proving

is

a search in the space of proofs