Relational Logic: Syntax & Semantic

Source: Computational Logic Lecture Notes
Stanford University

IF2121 Computational Logic 2023/2024

Informatics Engineering Study Program
School of Electrical Engineering and Informatics ITB

Contents

- Review
- Relational Logic → Introduction

Review

- Reasoning: information → conclusion
- Computational Logic
  - **Propositional Logic:**
    - Syntax → Simple sentence, Compound Sentence
    - **[something] is [anything]: [something] → [anything]**
    - Semantics → interpretation, evaluation, reverse evaluation, types of compound sentence
    - Logical Entailment :
      - Semantic Reasoning → Two tables, Validity Checking, Unsatisfiability Checking
      - Proof Method → Rules of Inference, Axiom Schemata, Propositional Resolution
  - **Relational Logic → Today**

Propositional Logic vs Relational Logic

- Constants refer to atomic propositions/ logical constants.

  raining *snowing* *wet*

- Compound sentences capture relationships among propositions

  $raining \vee snowing \Rightarrow wet$

- How to represent general concepts??

  In Propositional Logic:

  1. If Ali knows Budi, then Budi knows Ali (a → b)
  2. Ali knows Budi (a)
  3. Conclusion by Modus Ponens : Budi knows Ali (b)

  What if we want to say something more general, such as If person I knows person II, then person II knows person I

  In Relational Logic: $\forall x,y$ (knows(x,y) → knows(y,x))

Relational Logic Syntax

- Two new vocabularies:
  - Variables : begin with letters from the end of the alphabet
  - Constants:
    - *begin with either alphabetic letters (other than u,v, w, x, y, z),*
    - *mathematical characters (+, -, etc.),*
    - *or digits*
  - Example:
    - *u, v, w, x, y, z → ???*
    - *a, b, c, arthur, betty, cathy, 1, 2, ... →???*

Relational Logic Syntax (2): Constants

- Object constants refer to objects in the universe of discourse.
- Function constants denote functions.
  - father, mother, age, plus, times
- Relation constants refer to relations.
  - person, happy, parent, loves
- There is no syntactic distinction between object constants, function constants, and relation constants. The type of each such word is determined from context

Relational Logic Syntax (3): Arity

- The arity of a function constant or a relation constant is the number of arguments it takes.
  - Unary Function constants: *father1*, *mother*
  - Binary Function constants: *plus2*, *times2*
  - Ternary Function constants: *price3*
  - Unary Relation constants: *person1*, *happy1*
  - Binary Relation constants: *parent2*, *loves2*
  - Ternary Relation constants: *between3*
- The arity of a function constant or a relation constant is optionally notated as a subscript on the constant

Relational Logic Syntax (4): Term

- A term is either a variable, an object constant, or a functional term.
- Terms refer to items in the universe of discourse.
- Terms are analogous to noun phrases in natural language.

Functional Terms

- A functional term is an expression formed from an n-ary function constant and n terms enclosed in parentheses and separated by commas.
  - *father l (joe)*
  - *age l (joe)*
  - *plus2(x,2)*
- Functional terms are terms and, as such, can be nested.
  - *plus2(age l (father l (joe)),age l (mother l (joe)))*

Relational Logic Syntax (5): Sentences

- There are three types of sentences:
  - Relational sentences - analogous to the simple sentences in Propositional Logic
  - Logical sentences - analogous to the compound sentences in Propositional Logic
  - Quantified sentences - sentences that express the significance of variables

Relational Sentences

- A relational sentence is an expression formed from an n-ary relation constant and n terms enclosed in parentheses and separated by commas.
  - $happy_1(art)$
  - $loves_2(art,cathy)$
- Relational sentences are not terms and cannot be nested in terms or relational sentences.

  $happy_1(person_1(joe))$ - - > WRONG
  - Should be written:

  $happy_1(joe)$

  $person_1(joe)$

Logical Sentences

- Logical sentences in Relational Logic are analogous to those in Propositional Logic (Compound Sentences).
  - ¬loves(art,cathy)
  - (loves(art,betty) ∧ loves(betty,art))
  - (loves(art,betty) ∨ loves(art,cathy))
  - (loves(x,y) ⇒loves(y,x))
  - (loves(x,y) ⇐loves(y,x))
  - (loves(x,y)⇔loves(y,x))
- Parenthesization rules are the same as for Propositional Logic

Quantified Sentences

- Quantified sentences can be nested within other sentences.

  $\forall x.(apple(x) \vee \exists x.pear(x))$

  $\forall x.\forall y.loves(x,y)$

- Universally quantified sentences:
  - is used to assert that all objects have a certain property
  - $\forall x.p(x)$
- Existentially quantified sentences:
  - is used to assert that some object has a certain property
  - $\exists$ x.p(x)

Other Sentences

- A sentence is ground if and only if it contains no variables
  - Example:
  - Ground Sentence: *human(joe)*
  - Not Ground Sentence: ∀x.human(x)
- A sentence is open if and only if it has free variables (iff it is not in the scope of a quantifier of that variable)
  - Example:
  - *p(y)* → ∃*x.q(x, y)* - - > *Open sentence since y is a free variable*
  - ∀*y.(p(y)* → ∃*x.q(x, y))* - - > *Close sentence since x & y are bound*

Reminder

- Variables
- Constants: Object, [Functional, Relational] → has arguments (n-ary)
- Term: variables, object constants, functional terms → noun phrase in natural language
- Sentence: relational, logical, quantified
- Functional Term vs Relational Sentence:
  - Functional terms *may be used within other functional terms.*
  - Functional terms *may be used within relational sentences.*
  - Relational sentences may *not be used in functional terms.*
  - Relational sentences may *not be used in relational sentences.*

Natural Language Representation

- $\forall x.P(x)$
  For all x, x is P
- $\sim\forall x.P(x)$
  Not all x are P.
  Some x are P.
- $\forall x.\sim P(x)$
  All x are not P.

Natural Language Representation(2)

- $\exists x.P(x)$
  Some x is P
- $\exists x.\sim P(x)$
  Some x is not P
- $\forall x.\ \exists y.P(x,y)$
  For all x, there is y such that P

Natural Language Representation(3)

- $\forall x.(P(x) \rightarrow Q(x))$
  For all x, all P are Q
  All P is Q
- $\forall x.(P(x) \rightarrow \sim Q(x))$
  For all x, no P are Q
  No P is Q
- $\exists x.(P(x) \wedge Q(x))$
  For some x, x are P and Q
  Some P is Q
- $\exists x.(P(x) \wedge \sim Q(x))$
  For some x, x are P and not Q
  Some P is not Q

Examples

- All human is mortal.
  - If a thing is human, then it is mortal.

  $\forall x.(human(x) \rightarrow mortal(x))$
- Purple mushrooms are poisonous.
  - If a thing is a purple mushroom, then it is poisonous.
  - If a thing is mushroom and it is purple, then it is poisonous.

  $\forall x.(mushroom(x) \wedge purple(x) \rightarrow poisonous(x))$

Exercise 1

- Given:
  - Object Constants: art, betty, cathy
  - Functional Constants: $father_1$, $mother_1$, age_1, $plus_2$, $times_2$
  - Relational Constants: $person_1$, sad_1, $reflexive_1$, $parent_2$, $loves_2$, $friends_2$,
- Syntax Test:
  1. friends(father(art),betty)
  2. friends(mother(art),friends(father(art),betty))
  3. sad(person(cathy))
  4. loves(x,y) → loves(y,x)
  5. reflexive(z) → z(x,x)

Exercise 2

- Apakah ekpresi logika relasional di bawah ini legal atau tidak ? Jika tidak, jelaskan dimana letak kesalahannya dan mengapa, dengan ketentuan variabel dan konstanta sebagai berikut:
- *variables: x, y, z*
- *object constants: patrick, joe, kevinKW, PR, cemilan*
- *function constants: mother, anak, plus*
- *relational constants dengan aritas satu: hantu, biru, ramah, senang, sepupu*
- *relational constants dengan aritas dua: takutpada, suka, mengerjakan, teman*

Exercise 2 (2)

(a) ~suka(joe, PR) ∨ mengerjakan(joe, PR)

(b) plus(joe, z) → suka(mother(kevinKW), joe)

(c) hantu(x) ∧ ramah(x) → patrick(x)

(d) ~senang(z) ∧ takutpada(kevinKW, anak(sepupu(patrick)))

(e) biru(mother(anak(patrick))) ∨ ramah(anak(mother(kevinKW))) ⇔senang(joe)

(f) suka(joe, cemilan) ∧ suka(kevinKW, cemilan) → teman(kevinKW)

(g) takutpada(joe, sepupu(patrick))

(h) teman(patrick, joe) ∧ ~takutpada(kevinKW, hantu(patrick))

(i) suka(kevinKW, mother(x)) ∧ teman(kevinKW, x) → hantu(x) ∧ biru(y)

(j) plus(mother(anak(z)), anak(anak(anak(cemilan)))) ∧ plus(joe, kevinKW)

Relational Logic Semantics

- Review Propositional Logic Semantic:

  A Propositional logic *interpretation* is an association between the propositional constants in a propositional language and the truth values T or F

- The big question: what is a relational logic interpretation?
- There are no propositional constants, just object constants, function constants, and relation constants.
- To what do they refer?

Introduction to Relational Logic Semantic

- Modeling the World
  - Objects, Functions, Relations
  - Data
  - Models
- Semantics of Relational Logic
  - Atomic Sentences
  - Logical Sentences
  - Quantified Sentences

Exercise 3: Translate into FOL

Relation Constants:

- person(x)
- femur(x)
- leg(x)
- eye(x)
- part_of(x, y): x part of y
- has(x, y)
- heart(x)
- sinus_rhythm(x)
- seeing(x)
- living(x)
- rhythm(x)
- regular(x)
- differ(x,y): x and y are different

a) **All femurs are part of some legs**
b) **All living hearts have a rhythm**
c) **Not all eyes are seeing**
d) **All people have two eyes**

Exercise 4: Translate into FOL

Relation Constants:

- person(x)
- child(x)
- parent(x,y): x is the parent of y
- male(x)
- female(x)
- ancestor(x,y): x is the ancestor of y
- sibling(x,y)
- differ(x,y): x and y are different

a) **All people have two parents**
b) **No person is both male and female**
c) **All people have one male parent and one female parent**
d) **One child is a sibling of another if they both have the same two parents**

THANK YOU