Computational Linguistics Prep Course

Predicate Logic

Stefan Thater
Universität des Saarlandes
FR 4.7 Allgemeine Linguistik

Winter semester 2011/12

Outline

- Motivation: Natural language semantics
- First-order predicate logic
  - formal syntax
  - formal semantics
  - truth, validity, …
- Formalizing natural language expressions

Semantic Theory

A semantic theory should, amongst others, …

- provide **adequate semantic representations** that "capture" the meaning of natural language expressions
- provide mechanisms to compute semantic representations in a systematic way
- **explain semantic relations** between natural language sentences (equivalence, entailments, …)

Some Phenomena

Equivalence

(1) *A student did not pass [the exam]*

(2) *Not every student passed [the exam]*

Contradiction

(3) *A student did not pass*

(4) *Every student passed*

Entailment

(5) *John and Mary passed*

(6) *John passed*

Some Phenomena

Entailment: (1) $\models$ (2)

(1) *A blond student passed*

(2) *A student passed*

But: (3) $\not\models$ (4)

(3) *Every blond student passed*

(4) *Every student passed*

Some Phenomena

Entailment: (1), (2) $\models$ (3)

(1) *John is a blond student*

(2) *John is a tennis-player*

(3) *John is a blond tennis-player*

But: (4), (5) $\not\models$ (6)

(4) *John is a good student*

(5) *John is a tennis-player*

(6) *John is a good tennis-player*

Some Phenomena

(Structural) Ambiguity

(1) *John saw a man with a telescope*

(2) *Every student reads a book*

(3) *John seeks a unicorn*

(4) *Pola wants to marry a millionaire*

Semantics vs. Pragmatics

- We are mainly interested in the **literal meaning** of natural language expressions
- Although (1) somehow "suggests" (2), the entailment relation does not hold between the two sentences:

  (1) *John used to smoke 20 cigarettes a day few years ago*

  (2) *John does not smoke 20 cigarettes a day anymore*

Sense & Reference

Meaning is composed of sense and reference

- **Reference** = the object being referred to
- **Sense** = something that determines the reference

An Example: "rabbit"

- The reference is the set of rabbits
- The sense allows you to tell rabbits apart from non-rabits

Sentence Meaning

Referent of a sentence = truth value

- Some limitiations: questions, imperatives, performatives, “this statement is false”
- ⇒ we focus on declarative sentences

Sense of a sentence = conditions on truth

- To know the truth-conditions of a sentence is to know what the world has to be like for the sentence to be true.

Natural and formal languages

„There is in my opinion no important theoretical difference between natural languages and the artificial languages of logicians; indeed, I consider it possible to comprehend the syntax and semantics of both kinds of languages within a single natural and mathematically precise theory.“

Richard Montague (1970)

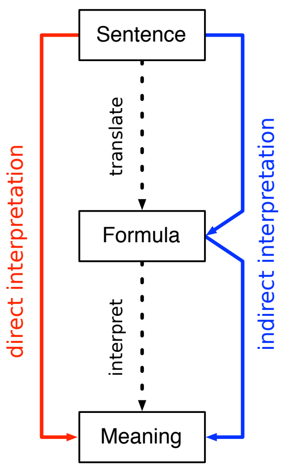
Direct vs. indirect interpretation

Indirect interpretation:

- Translate sentences into some appropriate logical representation language
- Interpret logical formulae

Direct interpretation:

- Interpret sentences directly (like a logical language)



Indirect Interpretation

(1) *Every student passed [the exam]*

Translation ("formalization")

- ∀x(student'(x) → pass'(x))

Interpretation

- ⟦∀x(student'(x) → pass'(x))⟧ = true iff ⟦student⟧ ⊆ ⟦pass⟧

Entailment

Entailment is a relation between sentences

- Strictily speaking: a relation between sentence meanings, i.e. the propositions expressed by the sentences

A sentence **A entails** a sentence **B** (A ⊨ B) iff whenever A is true, then B must also be true.

Entailment

A sentence **A entails** a sentence **B** (A ⊨ B) iff whenever A is true, then B must also be true.

(1) *Every student passed [the exam]*

- ∀x(student'(x) → pass'(x))
- ⟦∀x(…)⟧ = true iff ⟦student'⟧ ⊆ ⟦pass'⟧

(2) *Every blond student passed [the exam]*

- ∀x(blond'(x) ∧ student'(x) → pass'(x))
- ⟦∀x(…)⟧ = true iff ⟦blond'⟧ ∩ ⟦student'⟧ ⊆ ⟦pass'⟧

(1) ⊨ (2)

Textbooks

L.T.F. Gamut. Logic, Language and Meaning. Volume I: Introduction to Logic, University of Chicago Press, 1991.

Barbara H. Partee, Alice ter Meulen, Robert E. Wall. Mathematical Methods in Linguistics. Springer, 1990.

Predicate Logic

Predicate Logic

- Propositional logic talks about propositions (statements)
  - propositions have no internal structure (except connectives)
- Predicate logic decomposes simple statements into smaller parts:
  - predicates
  - terms
  - quantifiers

(1) *John works*
↦ work’(j)

(2) *John loves Mary*
↦ love’(j, m)

(3) *Everybody works*
↦ ∀x work’(x)

(4) *Somebody works*
↦ ∃x work’(x)

Predicate Logic – Vocabulary

- **Non-logical expressions:**
  - Individual constants: CON
  - n-place relation constants: $PRED^n$, for all $n \geq 0$
- **Infinite set of individual variables:** VAR

Predicate Logic – Syntax

- **Terms:** TERM = VAR ∪ CON
- **Atomic formulas:**
  - $R(t_1,\ldots, t_n)$ for $R \in PRED^n$ and $t_1, \ldots, t_n \in$ TERM
  - $t_1 = t_2$ for $t_1, t_2 \in$ TERM
- **Well-formed formulas:** the smallest set WFF such that
  - all atomic formulas are WFF
  - if φ and ψ are WFF, then $\neg\varphi$, $(\varphi \wedge \psi)$, $(\varphi \vee \psi)$, $(\varphi \rightarrow \psi)$, $(\varphi \leftrightarrow \psi)$ are WFF
  - if $x \in$ VAR, and φ is a WFF, then $\forall x\varphi$ and $\exists x\varphi$ are WFF

Quantification

∃x(…)

- “there is an x such that …”

∀x(…)

- “for every x it is the case that …”

Exercise – Translate into PL

(1) *John and Mary work*

↦ work’(j) ∧ work’(m)

(2) *A student works*

↦ ∃x(student’(x) ∧ work’(x))

(3) *A blond student works*

↦ ∃x(student’(x) ∧ blond’(x) ∧ work’(x))

(4) *A blond student works hard*

↦ ∃x(student’(x) ∧ blond’(x) ∧ work-hard’(x))

Exercise – Translate into PL

(1) *Mary loves a student*

↦ ∃x(student’(x) ∧ love’(m, x))

(2) *Every student works*

↦ ∀x (student’(x) → work’(x))

(3) *Nobody flunked*

↦ ¬∃x flunk’(x)

(4) *Barking dogs don’t bite*

↦ ∀x ((dog’(x) ∧ bark’(x)) → ¬bite’(x))

Scope

- If ∀xφ (∃xφ) is a subformula of a formula ψ, then φ is the **scope** of this occurrence of ∀x (∃x) in ψ.
- We distinguish distinct occurrences of quantifiers as there are formulae like ∀xA(x) ∧ ∀xB(x).
- Examples:
  - ∃x (∀y (T(y) ↔ x=y) ∧ F(x))
  - ∀x A(x) ∧ ∀x B(x)

Free and Bound Variables

- An *occurrence* of a variable x in a formula φ is **free in φ** if this occurrence of x does not fall within the scope of a quantifier ∀x or ∃x in φ.
- If ∀xψ (∃xψ) is a subformula of φ and x is free in ψ, then this occurrence of x is **bound by** this occurrence of the quantifier ∀x (∃x).
- Examples:
  - ∀x(A(x) ∧ B(x)) – x occurs bound in B(x)
  - ∀x A(x) ∧ B(x) – x occurs free in B(x)
- **A sentence** is a formula without free variables.

Predicate Logic – Semantics

- Expressions of Predicate Logic are interpreted relative to **model structures** and **variable assignments**.
- Model structures are our "mathematical picture" of the world. They provide interpretations for the non-logical symbols (predicate symbols, individual constants).
- Variable assignments provide interpretations for variables.

Model structures

- **Model structure:** $M = \langle U_M, V_M \rangle$
  - U_M is non-empty set – the "universe"
  - V_M is an interpretation function assigning individuals ($\in U_M$) to individual constants and n-ary relations over U_M to n-place predicate symbols:
    - $V_M(P) \subseteq U_M^n$ if P is an n-place predicate symbol
    - $V_M(c) \in U_M$ if c is an individual constant
- **Assignment function** for variables $g: VAR \rightarrow U_M$

Model structures – Example

$M = \langle U_M, V_M \rangle$

$U_M = \{ r_1, r_2, h_1, h_2 \}$

$V_M(\text{vincent}) = r_1$

$V_M(\text{mia}) = r_2$

$V_M(\text{rabbit}) = \{ r_1, r_2 \}$

$V_M(\text{white}) = \{ r_2 \}$

$V_M(\text{hat}) = \{ h_1, h_2 \}$

$V_M(\text{in}) = \{ (r_1, h_1) \}$