TDT4136 Introduction to Artificial Intelligence

Summary of Lecture 8: First Order Logic

Chapter 8 in the textbook

Liyuan Xing

Adjunct Associate Professor @ NAIL/IDI/NTNU
Machine Learning Engineer @ TrønderEnergi

2022

Outline

Syntax of FOL (Predicate) logic

Semantics of FOL

Quantifiers

Pros and Cons of propositional logic

PROs:

- Propositional logic is declarative: pieces of syntax correspond to facts
- Propositional logic allows partial information (unlike most data structures and databases)
- Propositional logic is compositional: meaning of $B_{1,1} \wedge P_{1,2}$ is derived from meaning of $B_{1,1}$ and of $P_{1,2}$
- Meaning in propositional logic is context-independent (unlike natural language, where meaning depends on context)

CON:

- Propositional logic has very limited expressive power (unlike natural language)

Limitations of propositional logic

- Some knowledge is hard or impossible to encode in the propositional logic.
- Two cases that are specially hard to represent:
  - Statements about similar objects, relations
  - Statements referring to groups of objects.

First Order Logic (FOL)

- More expressive than propositional logic. While PL assumes that the world contains facts, FOL assumes the world contains
  - Objects: trees, people, numbers, movies,Trump, maps, colours, hypotheses, Wumpus....
  - Relations: square, smelly, brother of, older than, owns, has colour, adjacent to....
  - Functions: brother-of, colour-of, adjacent-to,....
- Representing objects, their properties, relations and statements about them.

Syntax of FOL - elements

- Constants: NTNU, KingHarald, 5, ...
- Predicates: Brother, $>$, $=$, ...
- Functions: Sqrt, LeftLegOf
- Variables: x, y, a, b, ...
- Connectives: $\neg, \wedge, \vee, \rightarrow, \leftrightarrow$
- Quantifiers: $\forall, \exists$

- First order logic or "predicate calculus" introduces **variables** and **quantifiers** to refer to objects in the world, their relations, group of objects, and to express general rules.

Functions versus Relations

- Functions are a way of referring to individuals indirectly, e.g.,
  - brother-of (Janne) and Edvard would refer to the same object/individual if Janne's brother is the person named Edvard.
- Relations hold among objects
  - Brother(Janne,Edvard) is true if Edvard is Janne's brother
  - Unary, binary, n-ary relations

Syntax

- Atomic sentence: predicate(term$_1$,...,term$_n$)

  Term: constant, or variable or function(term$_1$,...,term$_m$)

- Complex sentences: Composed of atomic sentences using connectives

  $\neg S$, $S_1 \vee S_2$, $S_1 \wedge S_2$, $S_1 \implies S_2$, $S_1 \Leftrightarrow S_2$

  Examples:

  - Brother(JonSnow, AryaStark) $\implies$ Sister(AryaStark, JonSnow)
  - Childish(Trump) $\vee$ BestStudents(Students(NTNU), Norway)

Universal Quantifiers

- Quantifications express properties of collections of objects.
- $\forall$: "For all"
- $\forall \langle \textit{variable} \rangle \ \langle \textit{sentence} \rangle$
  We can state the following: $\forall$ x P(x)
  English translation: "for all values of x, P(x) is true"
  Example: P(x) : $x+1 \geq x$
  English translation: "for all values of x, $x+1 \geq x$ is true"

Universal quantification

- Everyone at NTNU is smart:
  $\forall x \;\; At(x, NTNU) \implies Smart(x)$
- $\forall x \;\; P$ is true in a model m iff P is true with x being each possible object in the model
- Equivalent to the conjunction of instantiations of P

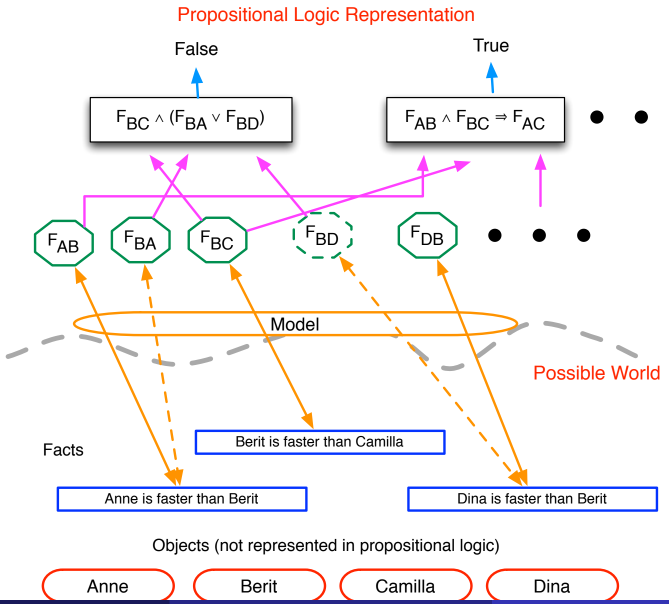
$$
\begin{aligned}
& (At(KingJohn, NTNU) \implies Smart(KingJohn)) \\
\wedge \quad & (At(Richard, NTNU) \implies Smart(Richard)) \\
\wedge \quad & (At(NTNU, NTNU) \implies Smart(NTNU)) \\
\wedge \quad & \ldots
\end{aligned}
$$

Existential quantification

- $\exists$: "There exist a/some"
- $\exists \langle variables \rangle \ \langle sentence \rangle$
- Someone at NTNU is smart:
  $\exists x \ \ At(x, NTNU) \land Smart(x)$
- $\exists x \ \ P$ is true in a model m iff P is true with x being **some** possible object in the model
- Equivalent to the disjunction of instantiations of P

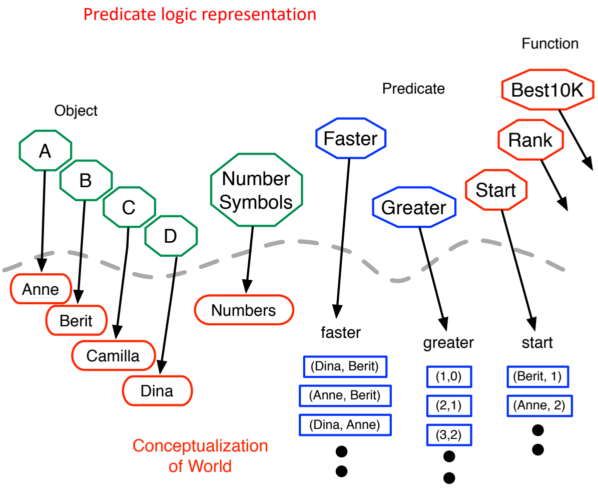
$$
\begin{array}{ll}
 & (At(KingJohn, NTNU) \land Smart(KingJohn)) \\
\lor & (At(Richard, NTNU) \land Smart(Richard)) \\
\lor & (At(NTNU, NTNU) \land Smart(NTNU)) \\
\lor & \ldots
\end{array}
$$

Example- Ski-race in Propositional Logic

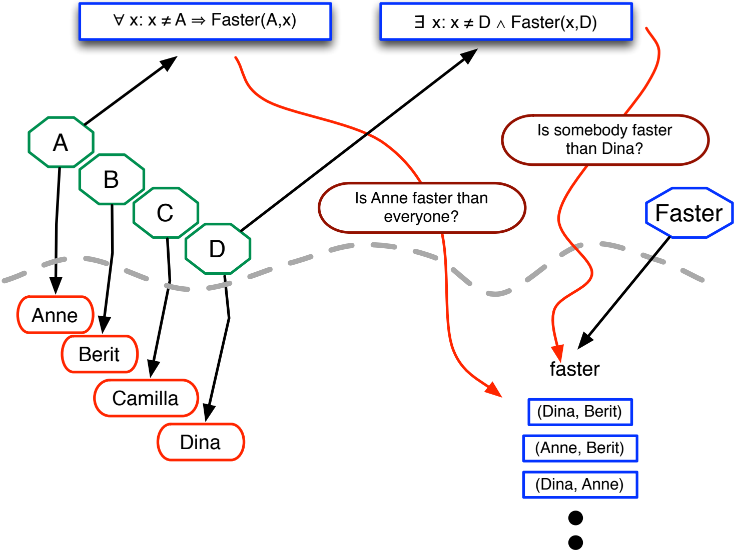


Example- Interpretations in First-Order Logic

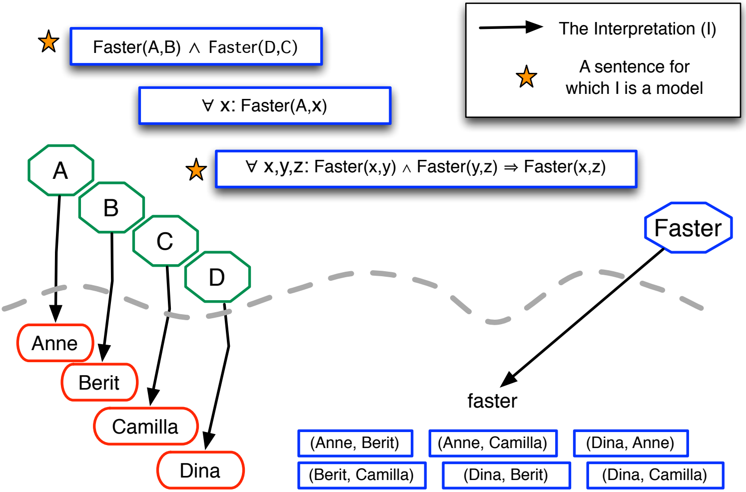
Interpretation = Mapping from constant, function and predicate symbols of the representation to the conceptualization.



Example - Evaluating Sentences with Quantified Variables



Example - Ski Models, Interpretation



Connections between $\forall$ and $\exists$

- All statements made with one quantifier can be converted into equivalent statements with the other quantifier by using negation.
- Remember De Morgan's Rule-
  - $P \wedge Q \equiv (\neg(\neg P \vee \neg Q))$
  - $P \vee Q \equiv (\neg(\neg P \wedge \neg Q))$
  - $\neg(P \wedge Q) \equiv \neg P \vee \neg Q$
  - $\neg(P \vee Q) \equiv \neg P \wedge \neg Q$
- Generalized De Morgan's rules
  - $\forall x\, P(x) \equiv \neg\exists x(\neg P(x))$
  - $\exists x\, P(x) \equiv \neg\forall x(\neg P(x))$
  - $\neg\forall x\, P(x) \equiv \exists x(\neg P(x))$
  - $\neg\exists x\, P(x) \equiv \forall x(\neg P(x))$

Multiple Quantifiers

- more complex sentences can be formulated by multiple variables and by nesting quantifiers
- variables must be introduced by quantifiers, and belong to the innermost quantifier that mention them
- examples
  $\forall x, y\ Parent(x, y) \implies Child(y, x)$
  $\forall x\ Human(x)\ \forall y\ Loves(x, y)$
  $\exists x\ Human(x)\ \forall y\ Loves(x, y)$
  $\exists x\ Human(x)\ \forall y\ Loves(y, x)$

Order of Quantifiers

- Reversing the order of the same type of quantifiers does not change the truth value of a sentence

  $\forall x\, \forall y P(x, y)$ is the same as $\forall y\, \forall x P(x, y)$

- $\exists x\, \forall y$ *and* $\forall x\, \exists y$ are not equivalent!

  $\forall x\, \exists y\ Loves(x, y)$
  $\exists y\, \forall x\ Loves(x, y)$

Negating multiple Quantifiers

- Recall negation rules for single quantifiers:

  $\neg\forall x\, P(x) = \exists x \neg P(x)$
  $\neg\exists\, x P(x) = \forall x \neg P(x)$

- You change the quantifier(s), and negate what it's quantifying:

  $\neg(\forall x\, \exists y\, P(x,y)) \equiv \exists x\, \neg\, \exists y P(x,y) \equiv \exists x\, \forall y\, \neg P(x,y)$