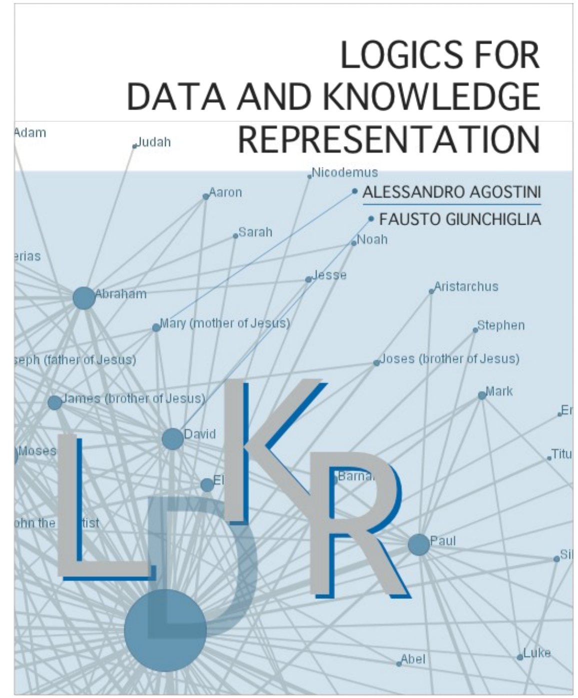


Logics for Data and Knowledge Representation

ClassL (part 1): syntax and semantics

Outline

- Syntax
  - Alphabet
  - Formation rules
- Semantics
  - Class-valuation
  - Venn diagrams
  - Satisfiability
  - Validity
- Reasoning
  - Comparing PL and ClassL
  - ClassL reasoning using DPLL

Language (Syntax)

- The syntax of ClassL is similar to PL
- Alphabet of symbols Σ_0

Σ_0

- **Descriptive**
  - **Constants**
    one proposition only
    A, B, C …
  - **Variables**
    they can be substituted by any proposition or formula
    P, Q, ψ …
- **Logical**
  $\sqcap$, $\sqcup$, $\neg$

- Auxiliary symbols: parentheses: ( )
- Defined symbols:

  $\bot$ (falsehood symbol, false, bottom) $\bot =_{\mathbf{df}} P \sqcap \neg P$

  T (truth symbol, true, top) $T =_{\mathbf{df}} \neg\bot$

Formation Rules (FR): well formed formulas

- Well formed formulas (wff) in ClassL can be described by the following BNF (*) grammar (codifying the rules):

  <Atomic Formula> ::= A | B | ... | P | Q | ... | ⊥ | ⊤

  <wff> ::= <Atomic Formula> | ¬<wff> | <wff> ⊓ <wff> | <wff> ⊔ <wff>

- Atomic formulas are also called atomic propositions
- Wff are class-propositional formulas (or just propositions)
- A formula is correct if and only if it is a wff

- Σ_0 + FR define a propositional language

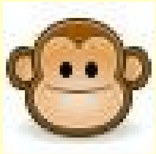
(*) BNF = Backus–Naur form (formal grammar)

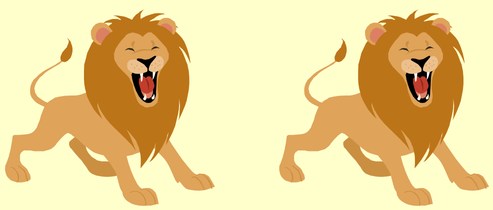
Extensional Semantics: Extensions

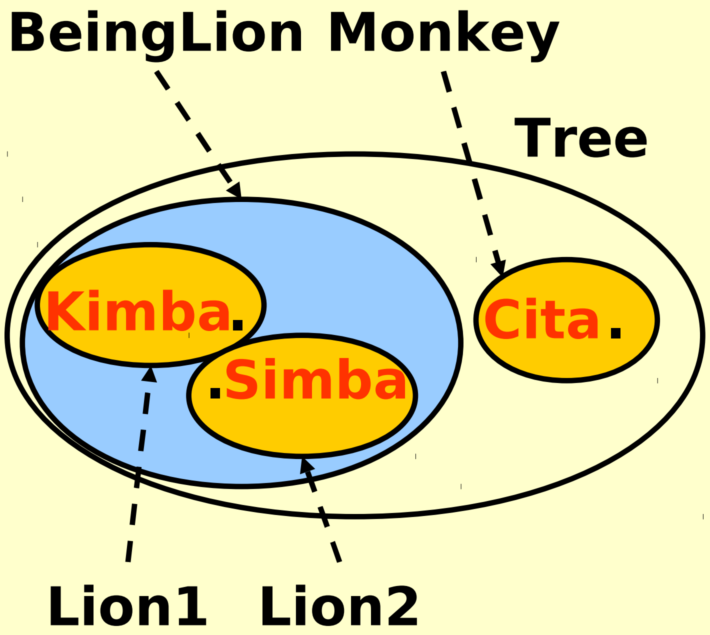
- The meanings which are intended to be attached to the symbols and propositions form the intended interpretation σ (sigma) of the language
- The semantics of a propositional language of classes L are extensional (semantics)
- The extensional semantics of L is based on the notion of "extension" of a formula (proposition) in L
- The extension of a proposition is the totality, or class, or set of all objects D (domain elements) to which the proposition applies

Extensional interpretation

D = {Cita, Kimba, Simba}







The World **The Mental Model** **The Formal Model**

Class-valuation σ

- In extensional semantics, the first central semantic notion is that of class-valuation (the interpretation function)
  - Given a Class Language L
  - Given a domain of interpretation U
  - A class valuation σ of a propositional language of classes L is a mapping (function) assigning to each formula ψ of L a set σ(ψ) of "objects" (truth-set) in U:

$$\sigma: L \rightarrow pow(U)$$

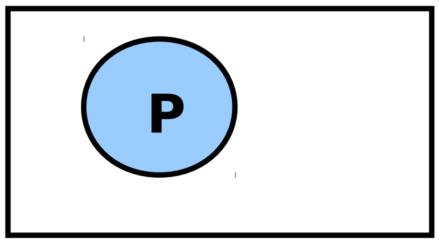
Class-valuation σ

- $\sigma(\bot) = \emptyset$
- $\sigma(\top) = U$ (Universal Class, or Universe)
- $\sigma(P) \subseteq U$, as defined by σ
- $\sigma(\neg P) = \{a \in U \mid a \notin \sigma(P)\} = \mathit{comp}(\sigma(P))$ (Complement)
- $\sigma(P \sqcap Q) = \sigma(P) \cap \sigma(Q)$ (Intersection)
- $\sigma(P \sqcup Q) = \sigma(P) \cup \sigma(Q)$ (Union)

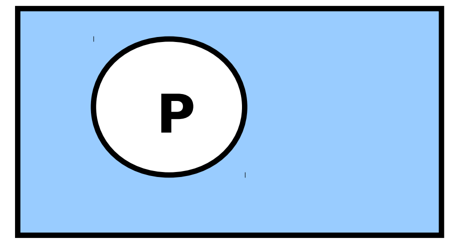
Venn Diagrams and Class-Values

- By regarding propositions as classes, it is very convenient to use Venn diagrams

$\sigma(P)$



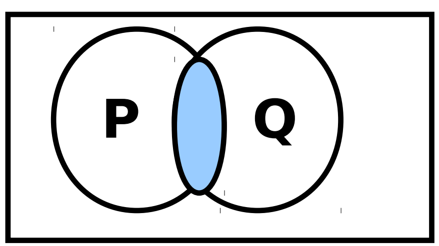
$\sigma(\neg P)$



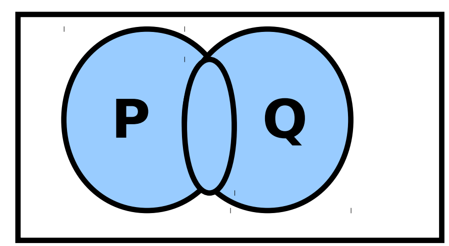
$\sigma(\bot)$

$\sigma(\top)$

$\sigma(P \sqcap Q)$



$\sigma(P \sqcup Q)$



Truth Relation (Satisfaction Relation)

- Let σ be a class-valuation on language L, we define the truth-relation (or class-satisfaction relation) $\models$ and write

$$\sigma \models P$$

(read: σ satisfies P) iff $\sigma(P) \neq \emptyset$

- Given a set of propositions Γ, we define

$$\sigma \models \Gamma$$

iff $\sigma \models \theta$ for all formulas $\theta \in \Gamma$

Model and Satisfiability

- Let σ be a class valuation on language L. σ is a model of a proposition P (set of propositions Γ) iff σ satisfies P (Γ).
- P (Γ) is class-satisfiable if there is a class valuation σ such that $\sigma \models P$ ($\sigma \models \Gamma$).

Truth, satisfiability and validity

- Let σ be a class valuation on language L.
- P is true under σ if P is satisfiable by σ (σ ⊨ P)
- P is valid if σ ⊨ P <u>for all</u> σ (notation: ⊨ P)
  In this case, P is called a tautology (always true)

NOTE: the notions of 'true' and 'false' are relative to some truth valuation.

NOTE: A proposition is true iff it is satisfiable

Reasoning on Class-Propositions

- Given a class-propositions P we want to reason about the following:
  - Model checking — Does σ satisfy P? (σ ⊨ P?)
  - Satisfiability — Is there any σ such that σ ⊨ P?
  - Unsatisfiability — Is it true that there are no σ satisfying P?
  - Validity — Is P a tautology? (true for all σ)

PL and ClassL are notational variants

- **Theorem**: P is satisfiable w.r.t. an intensional interpretation ν if and only if P is satifisfiable w.r.t. an extensional interpretation σ

	PL	**ClassL**
Syntax	$\wedge$	$\sqcap$
	$\vee$	$\sqcup$
	$\neg$	$\neg$
	$\top$	$\top$
	$\bot$	$\bot$
	P, Q…	P, Q…
Semantics	Δ={true, false}	Δ={o, …} (compare models)

ClassL reasoning using DPLL

- Given the theorem and the correspondences above, ClassL reasoning can be implemented using DPLL.
- The first step consists in translating P into P' expressed in PL

  - Model checking Does σ satisfy P? (σ ⊨ P?)
    Find the corresponding model ν and check that ν(P') = true
  - Satisfiability Is there any σ such that σ ⊨ P?
    Check that DPLL(P') succeeds and returns a ν
  - Unsatisfiability Is it true that there are no σ satisfying P?
    Check that DPLL(P') fails
  - Validity Is P a tautology? (true for all σ)
    Check that DPLL(¬P') fails