Lecture 10, Mar 19, 2024

Knowledge Representation and Reasoning (KR&R)

- How do we represent the information that we know, and reason intelligently using this information?
- This is the problem of *knowledge representation and reasoning* (KR&R)
  - *Representation*: symbolic encoding of propositions believed by some agent
    - Assigning symbols, and relating them
  - *Reasoning*: manipulation of said symbolic encoding to produce new propositions believed by the agent, but were not initially explicitly stated
    - Making rules for manipulating the symbols
- *Boolean algebra* or *propositional logic* is a set of rules for how to manipulate these propositions
  - In propositional logic, propositions are restricted to true statements

Propositional Logic

- For propositional logic, the set of operators are: $\vee$ (OR), $\wedge$ (AND), $\neg$ (NOT) and the brackets (and)
  - $\circ$ denotes any one of the binary operators
- Define PROP as the set of all propositions, OP as the set of all operators, and WDF (aka WFF) as the set of all well-defined (aka well-formed) formulas
  - Valid propositions follow a syntax; all propositions in the WDF is considered well-defined
  - If PROP is a finite set, then WDF is a countably infinite set
  - We construct WDF recursively as follows:
    - $\mathrm{FORM}_0 = \mathrm{PROP}$
    - $\mathrm{FORM}_{i+1} = \mathrm{FORM}_i \cup \{ (\alpha \circ \beta) \mid \alpha, \beta \in \mathrm{FORM}_i \} \cup \{ (\neg\alpha) \mid \alpha \in \mathrm{FORM}_i \}$
    - $\mathrm{FORM} = \displaystyle\bigcup_{i=0}^{\infty} \mathrm{FORM}_i$
  - Note that this excludes things like $(\alpha \circ \beta \circ \gamma)$
- Every formula is either:
  1. Atomic formula: a member of PROP
  2. Composite formula: made of a *primary connective* and set of *subformulas*
     - e.g. in $(\neg\alpha)$, $\neg$ is the primary connective and $\{ \alpha \}$ is the subformula; in $(\alpha \circ \beta)$, $\circ$ is the primary connective, and $\{ \alpha, \beta \}$ are the subformulas

Theorem

Unique Readability Theorem: Every (well-formed) formula is either atomic, or has a unique primary connective and unique set of well-defined subformulas.

- A *truth assignment* is a mapping $\tau \colon \mathrm{PROP} \mapsto \{ 0, 1 \}$
  - This basically assigns a value to all the propositions, but not all well-defined formulas
- The *extension* of τ for a formula φ is $\bar{\tau}(\varphi) = \begin{cases} \tau(p) & \varphi = p \in \mathrm{PROP} \\ \neg(\bar{\tau}(\alpha)) & \varphi = (\neg\alpha) \\ \circ(\bar{\tau}(\alpha), \bar{\tau}(\beta)) & \varphi = (\alpha \circ \beta) \end{cases}$
  - Sometimes we denote this as $\varphi(\tau)$
  - This assigns a value to all WDF
- We say that τ *models* φ, or $\tau \models \varphi$ iff $\bar{\tau}(\varphi) = 1$
- Let $\mathrm{AP}(\varphi) = \begin{cases} \{ p \} & \varphi = p \in \mathrm{PROP} \\ \mathrm{AP}(\alpha) & \varphi = (\neg\alpha) \\ \mathrm{AP}(\alpha) \cup \mathrm{AP}(\beta) & \varphi = (\alpha \circ \beta) \end{cases}$
  - This defines the set of atomic propositions that appear in a formula

Theorem

Relevance Lemma: If for all propositions p that appear in φ, both τ_1 and τ_2 assign it the same value, then τ_1 models φ iff τ_2 models φ, i.e.

$$\forall p \in \mathrm{AP}(\varphi), \tau_1(p) = \tau_2(p) \quad \Longrightarrow \quad \tau_1 \models \varphi \iff \tau_2 \models \varphi \text{ and } \varphi(\tau_1) = \varphi(\tau_2)$$

- Define $\varphi \models\dashv \psi$ iff $\forall \alpha \in 2^{\mathrm{PROP}}, \varphi(\tau) = \psi(\tau)$, i.e. they have the same value for all assignments
  - e.g. $((p \vee q) \vee p) \models\dashv (p \vee q)$
- $\mathrm{models}(\varphi) = \{\, \varphi \mid \tau \models \varphi \text{ or } \tau(\varphi) = 1 \text{ or } \bar{\tau}(\varphi) = 1 \,\}$
- *Valid* formulas, denoted $\models \varphi$, are formulas such that $\forall \tau, \tau \models \varphi$ – they are always true
  - If $\forall \tau, \tau \not\models \varphi$, then φ is *unsatisfiable* – they are always false
  - If $\exists \tau, \tau \models \varphi$, then φ is *satisfiable* – they are sometimes true
- Propositional logic is limited to only boolean variables, which makes cross-references between individuals in statements impossible
  - e.g. we can't model statements like "if x likes y and y plays golf then x watches golf", because we don't have variables or statements that are only sometimes true
- We also have no quantifiers; to state a property for all or some members of the domain, we have to explicitly list them