CS590U

Access Control: Theory and Practice

Lecture 14 (February 24)

Basics of Logic and Logic Programming

What is Logic?

- Formulas
- Syntactical approach
  - define how to derive new formulas from existing ones
  - $\Gamma \vdash \varphi$
- Semantic approach
  - define when a formula is a logical implication of other formulas
  - $\Gamma \models \varphi$

Example Logic Formulas

- From
  - $\forall X\ \forall Y$ (mother(X)$\wedge$child_of(Y,X) $\Rightarrow$loves(X,Y))
  - mother(mary)
  - child_of(tom,mary)
- Conclude
  - loves(mary,tom)

Kinds of Logic

- Propositional logic
  - classical, intuitionistic
- First order logic (predicate logic)
  - classical, intuitionistic
- Second order logic
- Modal logic

Propositional Logic

- $\wedge$ AND
- $\vee$ OR $p \vee q$ equivalent with $\neg(\neg p \wedge \neg q)$
- $\neg$ NOT
- $\Rightarrow$ $p \Rightarrow q$ equivalent with $\neg p \vee q$
- $\Leftrightarrow$ $p \Leftrightarrow q$ means $p \Rightarrow q \wedge q \Rightarrow p$

- Well formed formulas
  - a variable is a wff
  - $(\varphi \wedge \psi)$, $(\varphi \vee \psi)$, $(\neg\varphi)$, $(\varphi \Rightarrow \psi)$, $(\varphi \Leftrightarrow \psi)$ are wff's

Semantics of Propositional Logic

- A valuation of a formula φ is a truth assignment for every variable in φ
  - one can then evaluate φ
- A valuation of a formula φ is a model of φ if φ evaluates to true
- $\varphi_1, \ldots, \varphi_n \models \psi$ iff ψ is true in every model of $\varphi_1, \ldots, \varphi_n$
- A formula φ is satisfiable iff it has one model
- A formula φ is valid iff every valuation of it is a model for it

Conjunctive Normal Form

- Conjunctive Normal Form
- A formula is represented as conjunctions of disjunctions

- Checking validity of formulas in CNF is easy, but checking satisfiability is NP-complete

Horn Clauses

- A formula in CNF where each conjunct has 0 or 1 positive literal
  - p_1 — a fact
  - $p_1 \vee \neg p_2 \vee \ldots \vee \neg p_n$
    - i.e., $p_1 \Leftarrow p_2 \wedge \ldots \wedge p_n$ — a rule
  - $\neg p_1 \vee \neg p_2 \vee \ldots \vee \neg p_n$
    - i.e., $\Leftarrow p_2 \wedge \ldots \wedge p_n$ — a query
- Satisfiability of a formula in horn clauses can be decided in linear time

Predicate Logic

- An alphabet consists of predicates, constants, and variables
  - in $\forall X\ \forall Y\ (mother(X) \wedge child\_of(Y,X) \Rightarrow loves(X,Y)) \wedge mother(mary) \wedge child\_of(tom,mary)$
  - mother, child_of, loves are predicates
  - mary and tom are constants
  - X and Y are variables
  - $\forall$ and $\exists$ are quantifiers
- We ignore function symbols for this lecture

Closed Formulas (Sentences)

- Given a formula φ, the occurrence of a variable X is bound if it is inside the scope of a quantifier $\forall X$ or $\exists X$. Otherwise, the occurrence if free.
- A formula with no free occurrences of variables is said to be closed.
  - a closed formula is also known as a sentence.
- A formula with no variable is said to be ground.

Semantics of Predicate Logic

- A formula is just a string that can be parsed to a syntax tree
- A structure is a domain together with a set of relations (ignoring functions)
- An interpretation I of an alphabet maps
  - each constant to an element in the domain
  - each n-ary predicate to a relation

Semantics of Predicate Logic

- Given an interpretation I,
  - $\forall X\ \varphi$ evaluates to true iff for every element in I, φ with every free occurrence of X replaced by evaluates to true
  - $\exists X\ \varphi$ evaluates to true iff there exists an element in I, φ with every free occurrence of X replaced by evaluates to true
  - $p(c_1,c_2,\ldots,c_n)$ evaluates to true iff $(c_1,c_2,\ldots,c_n)$ is in the relation that p maps to

Models and Logical Consequence

- Given a set P of closed formulas, an interpretation I, I is a model of P iff every formula in P is true in I.
- A formula is unsatisfiable if it doesn't have a model
- Logical consequence: φ is a logical consequence of P iff φ is true in every model of P, written as $P \models \varphi$.
- Proving $P \models \varphi$ may be difficult, one way is to prove that $P \cup \neg\varphi$ is unsatisfiable

MSOf

Slide 13

MSOffice1 , 2/23/2005

Logical Inference

- Using rules to manipulate formulas to determine whether φ follows from P.
- Soundness and completeness

Logic Programming

- Rooted in Automated Theorem Proving
  - see an example
- The program consists of clauses
  - how to express grandchild in terms of child

Definite Clauses

- A definite clause has the form
  - $A_0 \leftarrow A_1 \wedge \ldots \wedge A_n$ where $n \geq 0$
  - When n=0, it is a fact
  - Otherwise, it is a rule
- Logical atoms, literals, clauses
- A define program is a finite set of definite clauses
- A definite goal has the form $\leftarrow A_1 \wedge \ldots \wedge A_n$
- The programmer has an intended model; the program describes features of the model. The programmer wants to know properties of the intended model
- The evaluation engine must be sound

The Least Herbrand Model

- Given an alphabet A, the Herbrand universe consists of all ground terms that can be constructed using symbols from A
  - when A doesn't contain any function symbols, the Herbrand universe is simply the set of all constants in A
- The Herbrand base consists of all ground atoms over A

Herbrand Model

- Herbrand interpretation
  - essentially a subset of the Herbran base, saying which ground atoms are true
- A Herbrand model of a program is a Herbrand interpretation such that every clause is true in it

Why Herbrand Model?

- Theorem: Let P be a definite program and G a definite goal, if $P \cup \{G\}$ has a model, then $P \cup \{G\}$ has a Herbrand model
- Corollary: if $P \cup \{G\}$ does not have a Herbrand model, then $P \models \neg G$.
- Thus, one only need to check whether G is false in all Herbrand models of P to determine whether $\neg$ G is true
- Theorem: Given two Herbrand model of a definite program, their intersection is also a Herbrand model

The Least Herbrand Model

- Theorem: There exists a unique least Herbrand model.
- Theorem: The least Herbrand model is the set of all ground atomic logical consequences of the program.

Construction of the Least Herbrand Model

- Using the immediate consequence operator
- The least fixpoint of the immediate consequence operator is the least Herbrand Model.

Next Lecture

- Overview of Trust Management