Discrete Mathematics

Inferencing by Resolution Refutation

Arijit Mondal
Dept of CSE
arijit@iitp.ac.in

Formulating Predicate Logic Statement

- **Wherever Mary goes, so does the lamb. Mary goes to school. So the lamb goes to school.**
  - **Predicate** $goes(x, y)$ **to denote** x **goes to** y
  - $F_1 : \forall x(goes(Mary, x) \rightarrow goes(Lamb, x))$
  - $F_2 : goes(Mary, School)$
  - $G : goes(Lamb, School)$
- **To prove:** $(F_1 \wedge F_2) \rightarrow G$ **is always true**

Variables and Predicate / Function Symbols

- **Variables, Free variables, Bound variables**
  - $\forall x(p(x, y))$
  - $\forall x\{p(x, y) \wedge \exists z\ q(x, y, z, w)\}$
  - $\forall x\{p(x, y) \wedge \exists z \exists y\ q(x, y, z, w)\}$
- **Symbols – proposition symbols, constant symbols, function symbols, predicate symbols**
- **Variables can be quantified in first order predicate logic**
- **Symbols cannot be quantified in first order predicate logic**
- **Interpretations are mapping of symbols to relevant aspects of a domain**

Terminology for Predicate Logic

- Domain: D
- Constant symbols: M, N, O, P, ...
- Variable symbols: x, y, z, ...
- Function symbols: F(x), G(x,y), ...
- Predicate symbols: p(x), q(x,y), ...
- Connectors: $\sim, \wedge, \vee, \rightarrow, \exists, \forall$
- Terms:
- Well-formed formula:
- Free and bound variables
- Interpretation, valid, non-valid, satisfiable, unsatisfiable

Resolution Refutation for Propositional Logic

- To prove validity of $M = ((F_1 \wedge F_2 \wedge \ldots \wedge F_n) \rightarrow G)$ we shall attempt to prove that $\neg M = (F_1 \wedge F_2 \wedge \ldots \wedge F_n \wedge \neg G)$ is unsatisfiable
- Steps for proof by resolution refutation
  - Convert to clausal form / Conjunctive Normal Form (CNF, Product of sums)
  - Generate new clauses using resolution rule
  - At the end, either false will be derived if the formula $\neg M$ is unsatisfiable implying M is valid

Resolution Refutation for Propositional Logic

- **If Asha is elected VP then Rajat is chosen as G-Sec and Bharati is chosen as Treasurer. Rajat is not chosen as G-Sec. Therefore Asha is not elected VP.**

Resolution Refutation for Propositional Logic

- **If Asha is elected VP then Rajat is chosen as G-Sec and Bharati is chosen as Treasurer. Rajat is not chosen as G-Sec. Therefore Asha is not elected VP.**
- F_1: $(a \rightarrow (b \wedge c)) = (\neg a \vee b) \wedge (\neg a \vee c)$
- F_2: $\neg b$, G: $\neg a$, $\neg G$: a
- **Let** $C_1 = a \vee b$, $C_2 = \neg a \vee c$, **then** $C_3 = b \vee c$ **can be derived**
- **To prove unsatisfiability use the resolution rule repeatedly to reach a situation where we have two contradictory clauses of the form** $C_1 = a$ **and** $C_2 = \neg a$ **from which false can be derived**
- **If the proposition formula is satisfiable then we will not reach a contradiction and eventually no new clauses will be derivable**
- **For propositional logic the procedure terminates**
- **Resolution rule is sound and complete**

Example

- **Rajesh either took the bus or came by cycle. If he came by cycle or walked to class he arrived late. Rajesh did not arrive late. Therefore he took the bus to class.**

Resolution Refutation for Predicate Logic

- Given a formula M which we wish to check for validity, we first check if there are any free variables. We then quantify all free variables universally
- Create $M' = \neg M$ and check for unsatisfiability of M'
- Steps:
  - Conversion to clausal form
    - Handling of variables and quantifiers, ground instances
  - Applying the resolution rule
    - Concept of unification
    - Principle of most general unifier (mgu)
    - Repeated application of resolution rule using mgu

Resolution Refutation for Predicate Logic

- Conversion to clausal form in predicate logic
  - Remove implications and other Boolean symbols converting to equivalent forms using $\neg$, $\vee$, $\wedge$
  - Move negates ($\neg$) inwards as close as possible
  - Standardize (rename) variables to make them unambiguous
  - Remove existential quantifiers by an appropriate new function / constant symbol taking into account the variables dependent on the quantifier (Skolemization)
  - Drop universal quantifiers
  - Distribute $\vee$ over $\wedge$ and convert to CNF

Conversion to clausal form

- Remove implications and other Boolean symbols converting to equivalent forms using $\neg, \vee, \wedge$
- Move negates ($\neg$) inwards as close as possible
- Standardize (rename) variables to make them unambiguous
- Remove existential quantifiers by an appropriate new function / constant symbol taking into account the variables dependent on the quantifier (Skolemization)
- Drop universal quantifiers
- Distribute $\vee$ over $\wedge$ and convert to CNF
- $\forall x\{(\forall y(student(y) \rightarrow likes(x, y))) \rightarrow (\exists z(likes(z, x)))\}$

Substitution, Unification, Resolution

- **Consider the following clauses:**
  - C_1: $\neg studies(x, y) \vee passes(x, y)$
  - C_2: $studies(Madan, z)$
  - C_3: $\neg passes(Chetan, Physics)$
  - C_4: $\neg passes(w, Mechanics)$
- **What new clauses can we derive by the resolution principle?**

Example

- F_1: $\forall x(contractor(x) \rightarrow \neg dependable(x))$
- F_2: $\exists x(engineer(x) \wedge contractor(x))$
- G: $\exists x(engineer(x) \wedge \neg dependable(x))$

Example

- F_1: $\forall x(dancer(x) \rightarrow graceful(x))$
- F_2: $student(Ayesha)$
- F_3: $dancer(Ayesha)$
- G: $\exists x(student(x) \wedge graceful(x))$

Thank you!