Inference in FOPL

04/02/2025

Koustav Rudra

Generalized Modus Ponens

- For atomic sentences $P_i, P_i', and\ q$, where there is a substitution θ such that
  - SUBST(θ, P_i') = SUBST(θ, P_i), for all i
  - $\frac{P_1', P_2', \ldots, P_n', (P_1 P_2 \ldots P_n \rightarrow q)}{SUBST(\theta, q)}$
  - Unification of P_1 with P_1', P_2 with P_2', …

Horn sentences

- Atomic sentences
  - perfect_sq(36)

- Implication with a conjunction of atomic sentences on the left and a single atom on the right
  - $\forall_{x,y}$ perfect_sq(x)∧prime(y)∧divides(x,y)→divides(x,square(y))

- No Existential quantifier

Modus Ponens - completeness

- Reasoning with Modus Ponens is incomplete

- Consider the example –
  - $\forall_x$ P(x) ➔ Q(x) $\forall_x$ ~P(x) ➔ R(x)
  - $\forall_x$ Q(x) ➔ S(x) $\forall_x$ R(x) ➔ S(x)

  - We should be able to conclude S(A)
  - The problem is that $\forall_x$ ~P(x) ➔ R(x) cannot be converted to Horn form and thus cannot be used by Modus Ponens

Godel Completeness Theorem

- For first-order logic, any sentence that is entailed by another set of sentences can be proved from that set
  - Godel did not suggest a proof procedure
  - In 1965 Robinson published his resolution algorithm

- Entailment in first-order logic is semi-decidable, that is,
  - we can show that sentences follow from premises if they do,
  - but we cannot always show if they do not.

Generalized Resolution Rule

- Generalized Resolution Rule:
- For atoms pi, qi, ri, si, where unify(p_j,q_k)= θ, we have

- $p_1 \land p_2 \land p_3 \ldots \land p_{n1} \rightarrow r_1 \lor r_2 \lor \ldots r_{n2}$
- $s_1 \land s_2 \land s_3 \ldots \land s_{n3} \rightarrow q_1 \lor q_2 \lor \ldots q_{n4}$

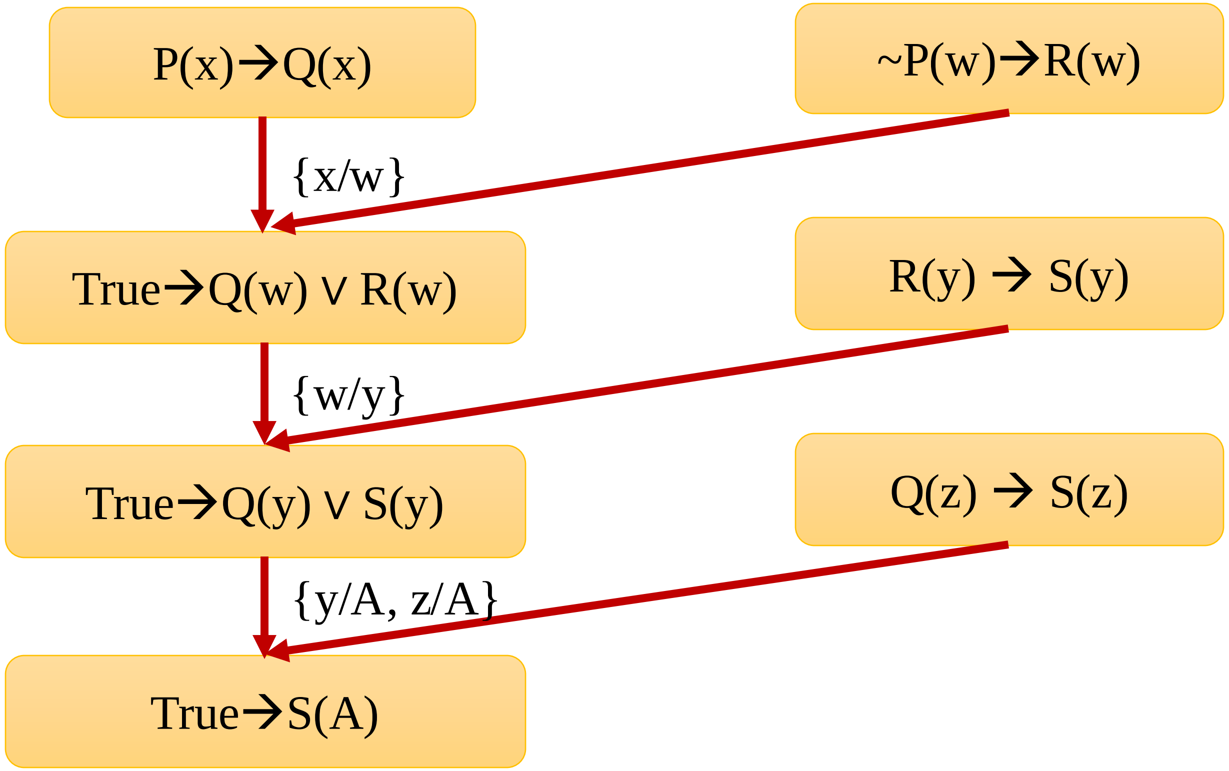
- SUBST(θ,
- $p_1 \land \ldots p_{j-1} \land p_{j+1} \ldots \land p_{n1} \land s_1 \land \ldots \land s_{n3} \rightarrow r_1 \lor \ldots r_{n2} \lor q_1 \lor \ldots q_{k-1} \lor q_{k+1} \ldots \lor q_{n4}$

Example

- $\forall_x\ P(x) \rightarrow Q(x)$
- $\forall_x\ Q(x) \rightarrow S(x)$

$\forall_x\ \sim P(x) \rightarrow R(x)$

$\forall_x\ R(x) \rightarrow S(x)$



How to automate the things?

Resolution in Predicate Logic

04/02/2025

Koustav Rudra

Conversion to Normal Form

- A formula is said to be in clause form if it is of the form:
  - $\forall_{x1}\forall_{x2}\ldots\ \forall_{xn}$[C1∧C2 ∧… ∧Ck]
  - Each of these clauses could be disjunction of some predicates

- All first-order logic formulas could be converted to clausal form

- $\forall_x$ {p(x)→ $\exists_z\{\sim\forall_y\left[q(x,y)\rightarrow p\left(f(x_1)\right)\right] \wedge \forall_y[q(x,y)\rightarrow p(x)]\}\}$

Conversion to Normal Form

- Step1: Take the existential closure and eliminate redundant quantifiers. This introduces $\exists_{x1}$ and eliminates $\exists_z$, so:

- $\forall_x$ {p(x)→ $\exists_z$ {$\sim\forall_y\left[q(x,y)\rightarrow p\left(f(x_1)\right)\right] \wedge \forall_y[q(x,y)\rightarrow p(x)]$}}

- $\exists_{x1}$ $\forall_x$ {p(x)→ {$\sim\forall_y\left[q(x,y)\rightarrow p\left(f(x_1)\right)\right] \wedge \forall_y[q(x,y)\rightarrow p(x)]$}}

Conversion to Normal Form

- Step2: Rename any variable that is quantified more than once
  - y has been quantified twice, so:

- $\exists_{x1}\ \forall_x\{\mathrm{p(x)} \rightarrow \{\sim\forall_y\left[q(x,y)\rightarrow p\left(f(x_1)\right)\right] \wedge \forall_y[q(x,y)\rightarrow p(x)]\}\}$

- $\exists_{x1}\ \forall_x\{\mathrm{p(x)} \rightarrow \{\sim\forall_y\left[q(x,y)\rightarrow p\left(f(x_1)\right)\right] \wedge \forall_z[q(x,z)\rightarrow p(x)]\}\}$

Conversion to Normal Form

- Step3: Eliminate implication

- $\exists_{x1} \forall_{x} \{p(x) \rightarrow \{\sim\forall_{y}[q(x,y) \rightarrow p(f(x_1))] \wedge \forall_{z}[q(x,z) \rightarrow p(x)]\}\}$

- $\exists_{x1} \forall_{x} \{\sim p(x) \vee \{\sim\forall_{y}[\sim q(x,y) \vee p(f(x_1))] \wedge \forall_{z}[\sim q(x,z) \vee p(x)]\}\}$

Conversion to Normal Form

- Step4: Move ~ all the way inwards

- $\exists_{x1}\ \forall_x\{\sim p(x) \lor \{\sim\forall_y[\sim q(x,y) \lor p(f(x_1))] \land \forall_z[\sim q(x,z) \lor p(x)]\}\}$

- $\exists_{x1}\ \forall_x\{\sim p(x) \lor \{\exists_y[q(x,y) \land \sim p(f(x_1))] \land \forall_z[\sim q(x,z) \lor p(x)]\}\}$

Conversion to Normal Form

- Step5: Push the quantifiers to the right

- $\exists_{x1}\ \forall_x \{\sim\mathrm{p(x)} \vee \{\exists_y \left[q(x,y) \wedge \sim p\left(f(x_1)\right)\right] \wedge \forall_z [\sim q(x,z) \vee p(x)]\}\}$

- $\exists_{x1}\ \forall_x \{\sim\mathrm{p(x)} \vee \{\left[\exists_y q(x,y) \wedge \sim p\left(f(x_1)\right)\right] \wedge [\forall_z \sim q(x,z) \vee p(x)]\}\}$

Conversion to Normal Form

- Step6: Eliminate existential quantifier (Skolemization)

- $\exists_{x1}\ \forall_x\{\sim p(x) \vee \{[\exists_y q(x,y) \wedge \sim p(f(x_1))] \wedge [\forall_z \sim q(x,z) \vee p(x)]\}\}$
- Pick out leftmost $\exists_y$B(y) and replace it by B(F(x_{i1}, $x_{i2,\ldots,}$ x_{in})), where:
  - x_{i1}, $x_{i2,\ldots,}$ x_{in} are all the distinct free variables of $\exists_y$B(y) that are universally quantified to the left of $\exists_y$B(y), and
  - F is any n-ary function constant which does not occur already

- $\forall_{x1}\forall_{x2}\forall_{x3}\exists_y$B(y, $x1$, $x2$, $x3$)
  - $\forall_{x1}\forall_{x2}\forall_{x3}$B(F($x1$, $x2$, $x3$), $x1$, $x2$, $x3$)
- $\forall_{x1}\forall_{x2}\forall_{x3}\exists_y$[B1(y, $x1$) $\wedge$ B2($x1, x2$)$\wedge$ B3($x2$, $x3$)]
  - $\forall_{x1}\forall_{x2}\forall_{x3}$ [B1(F($x1$, $x2$, $x3$), $x1$) $\wedge$ B2($x1, x2$)$\wedge$ B3($x2$, $x3$)]

Conversion to Normal Form

- Step6: Eliminate existential quantifier (Skolemization)

- $\exists_{x1}\ \forall_x \{\sim p(x) \lor \{[\exists_y q(x,y) \land \sim p(f(x_1))] \land [\forall_z \sim q(x,z) \lor p(x)]\}\}$

- $\forall_x \{\sim p(x) \lor \{[q(x,g(x)) \land \sim p(f(a))] \land [\forall_z \sim q(x,z) \lor p(x)]\}\}$

Conversion to Normal Form

- Step7: Move all universal quantifiers to the left

- $\forall_x\{\sim p(x) \vee \{[q(x, g(x)) \wedge \sim p(f(a))] \wedge [\forall_z \sim q(x,z) \vee p(x)]\}\}$

- $\forall_x \forall_z\{\sim p(x) \vee \{[q(x, g(x)) \wedge \sim p(f(a))] \wedge [\sim q(x,z) \vee p(x)]\}\}$

- Right side we have a set of predicates that are quantified from outside

Conversion to Normal Form

- Step8: Distribute ∧ over ∨
- $\forall_x \forall_z \{\sim p(x) \vee \{[q(x, g(x)) \wedge \sim p(f(a))] \wedge [\sim q(x, z) \vee p(x)]\}\}$
- $\forall_x \forall_z \{[\sim p(x) \vee [q(x, g(x)) \wedge \sim p(f(a))]] \wedge [\sim p(x) \vee [\sim q(x, z) \vee p(x)]]\}$
- $\forall_x \forall_z \{[\sim p(x) \vee q(x, g(x))] \wedge [\sim p(x) \vee \sim p(f(a))] \wedge [\sim p(x) \vee \sim q(x, z) \vee p(x)]\}$

- Right side we have a set of predicates that are quantified from outside
- Use Boolean algebra to get CNF
- Rename variables in different clauses so that same variable should not present in two clauses

Thank You