

Convertibility FOL into CNF

Input put order logic statement

Eliminate implications: replace $(A \rightarrow B)$ with $(\neg A \vee B)$

Move \neg (negation) inwards using De Morgan's law

Standardize variables ensure each quantifier have unique variables

move quantifiers to the front (prenex form)

Skolemize: eliminate existential quantifiers to introducing skolem function

Drop universal quantifiers

Distribute \vee over \wedge to obtain CNF form

Output CNF clauses

Output:

Original statement: $(A \wedge B) \rightarrow C$

CNF Form: $\neg A \vee \neg B \vee C$

creating a knowledge base using propositional
logic and proving using resolution.

Initialize knowledge base with propositional
logic statements
input query.

convert knowledge base and query into CNF.
Add query to CNF-clauses.

while True:

select two clauses from CNF-clauses
reduce the clauses to produce a new clause

If new clause is empty:

print "Query is proven using resolution"
break

If new clause is not already in CNF-clauses
add new clause to CNF-clauses

If no new clause can be generated:

print "Query cannot be proven using
resolution"

break

Output:

For knowledge base = ["A", "B", "A ~ B => C",
"C => D"]

query = "D"

Query is proven using resolution

Knowledge base with propositional logic

Initialize knowledge base with propositional
logic statements

Input query.

If forward-chaining (knowledge base, query)
print "Query is entailed to the knowledge
Base"

else:

print "Query is not entailed by the
knowledge Base"

Function forward-chaining (knowledge base, query)
Initialize agenda with known facts from
knowledge base

while agenda is not empty:

pop effect from agenda.

If fact matches query
return true

for each rule in knowledge-base

If fact satisfies a rule's premise
Add the rule's conclusion to agenda.

return failure.

Output:

For the knowledge-base = ["A", "B", "A & B => C",
"C => D"]

Query = "D"

Query is entailed to the knowledge base