

CS 480

Introduction to Artificial Intelligence

March 22, 2022

Announcements / Reminders

- **Programming Assignment #01:**
 - due: ~~March 13th~~ ~~March 20th~~ March 24th, 11:00 PM CST
- **New assignments will be posted soon**
- **New quiz: Monday**

- **Grading TA assignment:**
https://docs.google.com/spreadsheets/d/1Cav_GBTGC7fLGzXuBCAUmEuJYPeF-HMLCYvwPbq8Fus/edit?usp=sharing

Plan for Today

- **Predicate / First-Order Logic**

Predicate Logic Syntax: Summary

Predicate calculus symbols include:

- truth symbols: true and false
- terms represent specific objects in the world
 - constants, variables and functions
- predicate symbols refer to a particular relation between objects or represent facts
- function symbols refer to objects indirectly (via some relationship)
- quantifiers (\forall and \exists) and variables refer to collections of objects without explicitly naming each object

Proof by Resolution

The process of proving by resolution is as follows:

- A. Formalize the problem: “English to Predicate Logic”
- B. Derive $KB \wedge \neg Q$
- C. Convert $KB \wedge \neg Q$ into CNF (“standardized”) form
- D. Apply resolution rule to resulting clauses. New clauses will be generated (add them to the set if not already present)
- E. Repeat (C) until:
 - a. no new clause can be added (KB does NOT entail Q)
 - b. last two clauses resolve to yield the empty clause (KB entails Q)

Proof by Resolution

The process of proving by resolution is as follows:

- A. Formalize the problem: “English to Predicate Logic”
- B. Negate the input statement/claim C to obtain $\neg C$
- C. Convert $\neg C$ into CNF (“standardized”) form
- D. Apply resolution rule to resulting clauses. New clauses will be generated (add them to the set if not already present)
- E. Repeat (C) until:
 - a. no new clause can be added (C is false)
 - b. last two clauses resolve to yield the empty clause (C is true)

Predicate (First-Order) Logic to CNF

Variables and quantifiers are a challenge:

1. Eliminate all equivalences \Leftrightarrow and implications \Rightarrow
2. Reduce the scope of all \neg to single term (De Morgan)
3. Make all variable names unique (standardize apart)
4. Move quantifiers left (convert to prenex normal form)
5. Eliminate Existential quantifiers (skolemization)
6. Eliminate Universal quantifiers
7. Convert to conjunction of disjuncts
8. Create separate clause for each conjunct

1. Remove Equivalences/Implications

Use propositional logic laws to do it where possible.

Predicate (First-Order) Logic to CNF

Variables and quantifiers are a challenge:

1. Eliminate all equivalences \Leftrightarrow and implications \Rightarrow
2. Reduce the scope of all \neg to single term (De Morgan)
3. Make all variable names unique (standardize apart)
4. Move quantifiers left (convert to prenex normal form)
5. Eliminate Existential quantifiers (skolemization)
6. Eliminate Universal quantifiers
7. Convert to conjunction of disjuncts
8. Create separate clause for each conjunct

Predicate (First-Order) Logic to CNF

Variables and quantifiers are a challenge:

1. Eliminate all equivalences \Leftrightarrow and implications \Rightarrow
2. Reduce the scope of all \neg to single term (De Morgan)
3. Make all variable names unique (standardize apart)
4. Move quantifiers left (convert to prenex normal form)
5. Eliminate Existential quantifiers (skolemization)
6. Eliminate Universal quantifiers
7. Convert to conjunction of disjuncts
8. Create separate clause for each conjunct

2. Reduce the Scope of All \neg

Consider a predicate $N(x)$ asserting the fact that x is non-vegetarian. Now, let's create the following sentence:

$$\forall x (\neg N(x))$$

Which roughly translates to “**No one** is a non-vegetarian.” Let's try a slightly different sentence:

$$\neg[\forall x (N(x))]$$

Which roughly translates to “It is not true that **everyone** is a non-vegetarian”. This also means “**At least one** person is NOT a non-vegetarian” and we could rewrite it as:

$$\exists x (\neg N(x)), \text{ so } \neg[\forall x (N(x))] \equiv \exists x (\neg N(x))$$

Using the following logic, we can establish this equivalence:

$$\forall x (\neg N(x)) \equiv \neg[\exists x (N(x))]$$

2. Reduce the Scope of All \neg

Recall that for a domain of objects $\{a, b, c\}$, the sentence

$\forall x (N(x))$ is equivalent to $N(a) \wedge N(b) \wedge N(c)$

Similarly, the sentence equivalence:

$\exists x (N(x))$ is equivalent to $N(a) \vee N(b) \vee N(c)$

Now, if we apply De Morgan's rules, we get the following:

$$\neg[N(a) \vee N(b) \vee N(c)] \equiv [\neg N(a) \wedge \neg N(b) \wedge \neg N(c)]$$

and:

$$\neg[N(a) \wedge N(b) \wedge N(c)] \equiv [\neg N(a) \vee \neg N(b) \vee \neg N(c)]$$

2. Reduce the Scope of All \neg

Recall that for a domain of objects $\{a, b, c\}$, the sentence

$\forall x (N(x))$ is equivalent to $N(a) \wedge N(b) \wedge N(c)$

Similarly, the sentence equivalence:

$\exists x (N(x))$ is equivalent to $N(a) \vee N(b) \vee N(c)$

Now, if we apply De Morgan's rules, we get the following:

$$\frac{\neg[N(a) \vee N(b) \vee N(c)]}{\neg[\exists x (N(x))]} \equiv \frac{[\neg N(a) \wedge \neg N(b) \wedge \neg N(c)]}{\forall x (\neg N(x))}$$

and:

$$\frac{\neg[N(a) \wedge N(b) \wedge N(c)]}{\neg[\forall x (N(x))]} \equiv \frac{[\neg N(a) \vee \neg N(b) \vee \neg N(c)]}{\exists x (\neg N(x))}$$

Predicate (First-Order) Logic to CNF

Variables and quantifiers are a challenge:

1. Eliminate all equivalences \Leftrightarrow and implications \Rightarrow
2. Reduce the scope of all \neg to single term (De Morgan)
3. Make all variable names unique (standardize apart)
4. Move quantifiers left (convert to prenex normal form)
5. Eliminate Existential quantifiers (skolemization)
6. Eliminate Universal quantifiers
7. Convert to conjunction of disjuncts
8. Create separate clause for each conjunct

3. Make All Variable Names Unique

Given a qualified sentence:

$$\forall x (\text{crown}(x) \vee (\exists x \text{ brother}(\text{Richard}, x)))$$

change variables to avoid duplicates:

$$\forall x (\text{crown}(x) \vee (\exists z \text{ brother}(\text{Richard}, z)))$$

3. Make All Variable Names Unique

Given a qualified sentence:

$$\forall x (P(x) \Rightarrow Q(x)) \wedge \exists x (Q(x)) \wedge \exists z (P(z)) \wedge \exists x (Q(z) \Rightarrow R(z))$$

change variables to avoid duplicates:

$$\forall y (P(y) \Rightarrow Q(y)) \wedge \exists u (Q(u)) \wedge \exists w (P(w)) \wedge \exists z (Q(z) \Rightarrow R(z))$$

Also called: “standardizing variables apart”

Predicate (First-Order) Logic to CNF

Variables and quantifiers are a challenge:

1. Eliminate all equivalences \Leftrightarrow and implications \Rightarrow
2. Reduce the scope of all \neg to single term (De Morgan)
3. Make all variable names unique (standardize apart)
4. Move quantifiers left (convert to prenex normal form)
5. Eliminate Existential quantifiers (skolemization)
6. Eliminate Universal quantifiers
7. Convert to conjunction of disjuncts
8. Create separate clause for each conjunct

4. Move Quantifiers Left

Consider the following sentence in predicate logic:

$$\exists x (A(x) \vee \forall x (B(x)))$$

The two occurrences of x (x and x) do not refer to the same variable. Let's make all variables unique (standardize) first:

$$\exists x (A(x) \vee \forall y (B(y)))$$

Because variable y bound by $\forall y$ does not interact with the variable x bound by $\exists x$, we can extend the scope of the universal quantifier $\forall y$ to entire sentence:

$$\exists x \forall y (A(x) \vee (B(y)))$$

Now, $A(x) \vee (B(y))$ is **almost** a propositional logic sentence.

4. Move Quantifiers Left | PNF

A predicate logic formula φ is in **prenex normal form** (PNF) if it holds that:

- $\varphi = Q_1x_1 \dots Q_nx_n \psi$
- ψ is a quantifierless sentence
- $Q_i \in \{\forall, \exists\}$ for $i = 1, \dots, n$

For example this sentence is NOT in PNF:

$$\exists x (A(x) \vee \forall y (B(y)))$$

This sentence is in PNF:

$$\exists x \forall y (A(x) \vee (B(y)))$$

4. Move Quantifiers Left | PNF

Every predicate logic sentence can be transformed into an equivalent sentence in prenex normal form.

Predicate (First-Order) Logic to CNF

Variables and quantifiers are a challenge:

1. Eliminate all equivalences \Leftrightarrow and implications \Rightarrow
2. Reduce the scope of all \neg to single term (De Morgan)
3. Make all variable names unique (standardize apart)
4. Move quantifiers left (convert to prenex normal form)
5. Eliminate Existential quantifiers (skolemization)
6. Eliminate Universal quantifiers
7. Convert to conjunction of disjuncts
8. Create separate clause for each conjunct

5. Eliminating Existential Quantifiers

In order to convert a predicate logic to a propositional logic CNF form we need to remove quantifiers.

Existential quantifiers can appear in sentences:

- in isolation (\exists is **OUTSIDE** the scope of some \forall):

$$\exists x (A(x))$$

This can be resolved using **Skolem constant(s)**.

- in relation (\exists is **INSIDE** the scope of some \forall)

$$\forall y (\exists x (A(x, y)))$$

This can be resolved using **Skolem function(s)**.

The process is known as **skolemization**.

5. Eliminating Existential Quantifiers

Any object **variable** that is existentially quantified outside of the scope of a universal quantifier, such as

$$\exists x (A(x))$$

can be replaced by a single new constant expression $A(t)$, where t is a **Skolem constant**:

$$\exists x (A(t))$$

and the existential quantifier can be dropped to obtain

$$A(t)$$

5. Eliminating Existential Quantifiers

With multiple **variables** that are existentially quantified outside of the scope of a universal quantifier, such as

$$\exists x, y (A(x, y))$$

multiple corresponding **Skolem constants** (**t**, **u**) will be needed to create a new constant expression $A(t, u)$,

$$\exists x, y (A(t, u))$$

and the existential quantifier can be dropped to obtain

$$A(t, u)$$

5. Eliminating Existential Quantifiers

With multiple **variables** bound to **different existential quantifiers** outside of the scope of a universal quantifier:

$$[\exists x (B(x))] \vee [\exists y (C(y))]$$

multiple **Skolem constants** **t**, **u** will be needed to create new constant expressions **B(t)** and **C(u)**:

$$[\exists x (B(t))] \vee [\exists y (C(u))]$$

and existential quantifiers can be dropped to obtain

$$[B(t)] \vee [C(u)]$$

5. Eliminating Existential Quantifiers

Any object **variable** that is existentially quantified outside of the scope of a universal quantifier, such as

$$[\forall x (B(x))] \vee [\exists y (C(y))]$$

can be replaced by a single new constant expression $C(t)$, where t is a **Skolem constant**:

$$[\forall x (B(x))] \vee [\exists y (C(t))]$$

and the existential quantifier can be dropped to obtain

$$[\forall x (B(x))] \vee [(C(t))]$$

5. Eliminating Existential Quantifiers

We can say that following predicate logic sentences:

$$\exists x (A(x)) \equiv_I A(t)$$

$$\exists x, y (A(x, y)) \equiv_I A(t, u)$$

$$[\exists x (B(x))] \vee [\exists y (C(y))] \equiv_I [B(t)] \vee [C(u)]$$

$$[\forall x (B(x))] \vee [\exists y (C(y))] \equiv_I [\forall x (B(x))] \vee [C(t)]$$

are inferentially equivalent (\equiv_I). Skolemization leads to sentences that are **not completely equivalent**, but this is **good enough for proofs and inference**.

5. Eliminating Existential Quantifiers

Inferentially equivalent sentences are **not completely equivalent**, but this is **good enough for proofs**. Why?

Consider following two predicate logic sentences:

$\exists x$ (**studies**(x)): there exist **at least one** x who studies
studies(t): t studies (**just one, specific** object t)

Constant t is assumed to be a possible value for **variable** x .
If for some object t , **studies**(t) is true, then $\exists x$ (**studies**(x))
also must be true (t and possibly other objects **study**).

5. Eliminating Existential Quantifiers

Note: when choosing a Skolem constant for a existentially quantified expressions such as:

$$\exists x (A(x))$$

DON'T choose EXISTING constants as Skolem constants to create a new constant expression $A(t)$, where t is a Skolem constant:

So: $\exists x (A(t))$ YES,

but $\exists x (A(\text{lukeSkywalker}))$ NO

Assuming that lukeSkywalker is an existing object.

5. Eliminating Existential Quantifiers

Any object **variable** that is existentially quantified inside of the scope of a universal quantifier, such as:

$$\forall y (\exists x (A(x, y)))$$

can be replaced by with a **Skolem function** of the universal variable $f(y)$:

$$\forall y (\exists x (A(f(y), y)))$$

and the existential quantifier can be dropped to obtain

$$\forall y (A(f(y), y))$$

5. Eliminating Existential Quantifiers

An existential quantifier inside of the scope of MORE THAN ONE universal quantifier, such as:

$$\forall y \forall z (\exists x (B(x, y, z)))$$

can be replaced by with a multivariable **Skolem function** of $g(y, z)$:

$$\forall y \forall z (\exists x (B(g(y, z), y, z)))$$

and the existential quantifier can be dropped to obtain

$$\forall y \forall z (B(g(y, z), y, z))$$

5. Eliminating Existential Quantifiers

Consider the following example:

$$\forall x [\exists y (A(x) \Rightarrow B(y)) \vee \forall w (\exists z (D(x) \wedge E(w) \wedge F(z) \Rightarrow C(z)))]$$

can be modified using skolemization to obtain:

$$\forall x [(A(x) \Rightarrow B(f(x))) \vee \forall w ((D(x) \wedge E(w) \wedge F(g(x, w)) \Rightarrow C(g(x, w)))]$$


Skolem functions $f()$ and $g()$.

Variable y is inside the scope of $\forall x$, hence: $f(x)$

Variable z is inside the scope of $\forall x$ and $\forall w$, hence: $g(x, w)$

5. Eliminating Existential Quantifiers

In general: existential quantifiers can also be eliminated through the use of **Existential Instantiation**.

For any sentence S , variable x , and constant symbol k (that does not appear anywhere in the knowledge base), we can use the following rule:

$$\frac{\exists x S}{\text{SUBST}(\{x / k\}, S)}$$

Where is a result of applying substitution $\{x / k\}$ to the sentence S .

5. Eliminating Existential Quantifiers

For example, from the sentence:

$$\exists x (\text{crown}(x) \wedge \text{onHead}(x, \text{John}))$$

we can infer the sentence

$$\text{crown}(C_1) \wedge \text{onHead}(C_1, \text{John})$$

using the substitution $\{x / C_1\}$ as long as C_1 does not exist in the knowledge base.

Predicate (First-Order) Logic to CNF

Variables and quantifiers are a challenge:

1. Eliminate all equivalences \Leftrightarrow and implications \Rightarrow
2. Reduce the scope of all \neg to single term (De Morgan)
3. Make all variable names unique (standardize apart)
4. Move quantifiers left (convert to prenex normal form)
5. Eliminate Existential quantifiers (skolemization)
6. Eliminate Universal quantifiers
7. Convert to conjunction of disjuncts
8. Create separate clause for each conjunct

6. Eliminating Universal Quantifiers

In general universal quantifiers can also be eliminated through the use of **Universal Instantiation**.

For any sentence S , variable x , and constant symbol g (that does not appear anywhere in the knowledge base), we can use the following rule:

$$\frac{\forall x S}{\text{SUBST}(\{x / g\}, S)}$$

Where is a result of applying substitution $\{x / g\}$ to the sentence S .

6. Eliminating Universal Quantifiers

For example, from the sentence:

$$\forall x (\text{king}(x) \wedge \text{greedy}(x) \Rightarrow \text{evil}(x))$$

we can infer the sentence

$$\text{king}(\text{John}) \wedge \text{greedy}(\text{John}) \Rightarrow \text{evil}(\text{John})$$

using the substitution $\{x / \text{John}\}$.

Problem: Argument Mismatch

The goal of the resolution process is to find a contradiction. When two contradicting literals “cancel” each other out, we can end up with an empty clause:

$$\frac{(\textcolor{blue}{w}), (\neg \textcolor{blue}{w})}{()}$$

It is not so easy in predicate logic. This

$$\frac{(\textcolor{green}{setting}(\textcolor{blue}{sun})), (\neg \textcolor{green}{setting}(\textcolor{blue}{sun}))}{()}$$

will work (**predicate** arguments match). This

$$\frac{(\textcolor{green}{beautiful}(\textcolor{blue}{day})), (\neg \textcolor{green}{beautiful}(\textcolor{blue}{night}))}{??????}$$

will not, because **predicate** arguments don't match.

Predicate (First-Order) Logic to CNF

Variables and quantifiers are a challenge:

1. Eliminate all equivalences \Leftrightarrow and implications \Rightarrow
2. Reduce the scope of all \neg to single term (De Morgan)
3. Make all variable names unique (standardize apart)
4. Move quantifiers left (convert to prenex normal form)
5. Eliminate Existential quantifiers (skolemization)
6. Eliminate Universal quantifiers
7. Convert to conjunction of disjuncts
8. Create separate clause for each conjunct

Universal Quantifier: Conjunctions

Universal quantifier (“for all”) indicates that a sentence is true for **all possible values of the variable**. For example:

$$\forall x \text{ likes}(x, \text{cake})$$

is true if $\text{likes}(x, \text{cake})$ is true **for all interpretations** of variable x . Assuming that

$$x \in \{x_1, x_2, \dots, x_n\}$$

we can rewrite $\forall x \text{ likes}(x, \text{cake})$ as:

$$\text{likes}(x_1, \text{cake}) \wedge \text{likes}(x_2, \text{cake}) \wedge \dots \wedge \text{likes}(x_n, \text{cake})$$

Eliminating Universal Quantifiers

In general universal quantifiers can also be eliminated through the use of **Universal Instantiation**.

For any sentence S , variable x , and constant symbol g (that does not appear anywhere in the knowledge base), we can use the following rule:

$$\frac{\forall x S}{\text{SUBST}(\{x / g\}, S)}$$

Where is a result of applying substitution $\{x / g\}$ to the sentence S .

Eliminating Universal Quantifiers

For example, from the sentence:

$$\forall x (\text{king}(x) \wedge \text{greedy}(x) \Rightarrow \text{evil}(x))$$

we can infer the sentence

$$\text{king}(\text{John}) \wedge \text{greedy}(\text{John}) \Rightarrow \text{evil}(\text{John})$$

using the substitution $\{x / \text{John}\}$.

Eliminating Existential Quantifiers

In general: existential quantifiers can also be eliminated through the use of **Existential Instantiation**.

For any sentence S , variable x , and constant symbol k (that does not appear anywhere in the knowledge base), we can use the following rule:

$$\frac{\exists x S}{\text{SUBST}(\{x / k\}, S)}$$

Where is a result of applying substitution $\{x / k\}$ to the sentence S .

Propositionalization

The idea:

- Replace an existentially quantified sentence with ONE instantiation (Skolemization)
- Replace an universally quantified sentence with ALL POSSIBLE instantiations

For example, from the sentence:

$$\forall x (\text{king}(x) \wedge \text{greedy}(x) \Rightarrow \text{evil}(x))$$

Assume: there are TWO possible values/objects for x : {John, Richard}. We obtain:

$$(\text{king}(\text{John}) \wedge \text{greedy}(\text{John}) \Rightarrow \text{evil}(\text{John}))$$

$$(\text{king}(\text{Richard}) \wedge \text{greedy}(\text{Richard}) \Rightarrow \text{evil}(\text{Richard}))$$

Propositionalization

Now, we can continue the conversion of:

$$(\text{king}(\text{John}) \wedge \text{greedy}(\text{John}) \Rightarrow \text{evil}(\text{John}))$$

$$(\text{king}(\text{Richard}) \wedge \text{greedy}(\text{Richard}) \Rightarrow \text{evil}(\text{Richard}))$$

by replacing each atomic predicate logic symbol with a propositional logic symbol

$$(\text{JohnIsKing} \wedge \text{JohnIsGreedy} \Rightarrow \text{JohnIsEvil})$$

$$(\text{RichardIsKing} \wedge \text{RichardIsGreedy} \Rightarrow \text{RichardIsEvil})$$

Can you see potential problems?

Propositionalization

What if, in addition to:

$$(\text{king}(\text{John}) \wedge \text{greedy}(\text{John}) \Rightarrow \text{evil}(\text{John}))$$

$$(\text{king}(\text{Richard}) \wedge \text{greedy}(\text{Richard}) \Rightarrow \text{evil}(\text{Richard}))$$

we also had a function $\text{Father}(\cdot)$?

You can easily end up with infinite nesting of the following nature:

$$\text{Father}(\text{Father}(\text{Father}(\text{Father}(\text{John}))))$$

That leads to an infinite number of clauses!

Unification

Predicate logic inference rules **require finding substitutions that make two different logical expressions look identical.**

The process is called **unification**. A UNIFY algorithm takes **two sentences** p and q and returns a unifier θ for them (a substitution) if one exists:

$$\text{UNIFY}(p, q) = \theta, \text{ where } \text{SUBST}(\theta, p) = \text{SUBST}(\theta, q)$$

Unification: Examples

$\text{UNIFY}(\text{sentenceA}, \text{sentenceB}) = \{\text{unifier for sentenceA and sentenceB}\}$

$\text{UNIFY}(\mathbf{p}, \mathbf{q}) = \{\theta\}$

$\text{UNIFY}(\mathbf{p}, \mathbf{q}) = \{\text{variable / unifying value}\}$

Examples:

$\text{UNIFY}(\text{Knows}(\text{John}, \mathbf{x}), \text{Knows}(\text{John}, \text{Jane})) = \{\mathbf{x}/\text{Jane}\}$

$\text{UNIFY}(\text{Knows}(\text{John}, \mathbf{x}), \text{Knows}(\mathbf{y}, \text{Bill})) = \{\mathbf{x}/\text{Bill}, \mathbf{y}/\text{John}\}$

$\text{UNIFY}(\text{Knows}(\text{John}, \mathbf{x}), \text{Knows}(\mathbf{y}, \text{Mother}(\mathbf{y}))) = \{\mathbf{x}/\text{Mother}(\text{John}), \mathbf{y}/\text{John}\}$

$\text{UNIFY}(\text{Knows}(\text{John}, \mathbf{x}), \text{Knows}(\mathbf{x}, \text{Elizabeth})) = \text{failure}$
($\{\mathbf{x}/\text{John}, \mathbf{x}/\text{Elizabeth}\}$ is not possible)

Most General Unifier (MGU)

But.... there can be multiple unifiers for a pair of sentences. Which one to choose?

Every UNIFIABLE pair of sentences has a SINGLE **most general unifier** that is unique.

UNIFY algorithm will find MGU.

Unification

function UNIFY($x, y, \theta = \text{empty}$) **returns** a substitution to make x and y identical, or *failure*
 if $\theta = \text{failure}$ **then return** *failure*
 else if $x = y$ **then return** θ
 else if VARIABLE?(x) **then return** UNIFY-VAR(x, y, θ)
 else if VARIABLE?(y) **then return** UNIFY-VAR(y, x, θ)
 else if COMPOUND?(x) **and** COMPOUND?(y) **then**
 return UNIFY(ARGS(x), ARGS(y), UNIFY(OP(x), OP(y), θ))
 else if LIST?(x) **and** LIST?(y) **then**
 return UNIFY(REST(x), REST(y), UNIFY(FIRST(x), FIRST(y), θ))
 else return *failure*

function UNIFY-VAR(var, x, θ) **returns** a substitution
 if $\{var/val\} \in \theta$ for some val **then return** UNIFY(val, x, θ)
 else if $\{x/val\} \in \theta$ for some val **then return** UNIFY(var, val, θ)
 else if OCCUR-CHECK?(var, x) **then return** *failure*
 else return add $\{var/x\}$ to θ

Converting FOL to CNF: Example 1

Original sentence S:

$$\forall w ([P_1(w) \vee P_2(w) \Rightarrow P_3(w)] \vee [\exists x (\exists y (P_6(x, y) \Rightarrow P_4(w, x)))]) \wedge [\forall w (P_5(w))]$$

Predicate (First-Order) Logic to CNF

Variables and quantifiers are a challenge:

1. **Eliminate all equivalences \Leftrightarrow and implications \Rightarrow**
2. Reduce the scope of all \neg to single term (De Morgan)
3. Make all variable names unique (standardize apart)
4. Move quantifiers left (convert to prenex normal form)
5. Eliminate Existential quantifiers (skolemization)
6. Eliminate Universal quantifiers
7. Convert to conjunction of disjuncts
8. Create separate clause for each conjunct

Converting FOL to CNF: Example 1

Original sentence S:

$$\forall w ([P_1(w) \vee P_2(w) \Rightarrow P_3(w)] \vee [\exists x (\exists y (P_6(x, y) \Rightarrow P_4(w, x)))] \wedge [\forall w (P_5(w))])$$

By Implication Law ($p \Rightarrow q \equiv \neg p \vee q$):

$$\forall w ([\underline{P_1(w)} \vee \underline{P_2(w)} \Rightarrow \underline{P_3(w)}] \vee [\exists x (\exists y (P_6(x, y) \Rightarrow P_4(w, x)))] \wedge [\forall w (P_5(w))])$$

becomes:

$$\forall w ([\neg(\underline{P_1(w)} \vee \underline{P_2(w)}) \vee \underline{P_3(w)}] \vee [\exists x (\exists y (P_6(x, y) \Rightarrow P_4(w, x)))] \wedge [\forall w (P_5(w))])$$

By Implication Law ($p \Rightarrow q \equiv \neg p \vee q$):

$$\forall w ([\neg(P_1(w) \vee P_2(w)) \vee P_3(w)] \vee [\exists x (\exists y (\underline{P_6(x, y)} \Rightarrow \underline{P_4(w, x)}))] \wedge [\forall w (P_5(w))])$$

becomes:

$$\forall w ([\neg(P_1(w) \vee P_2(w)) \vee P_3(w)] \vee [\exists x (\exists y (\neg \underline{P_6(x, y)} \vee \underline{P_4(w, x)}))] \wedge [\forall w (P_5(w))])$$

Predicate (First-Order) Logic to CNF

Variables and quantifiers are a challenge:

1. Eliminate all equivalences \Leftrightarrow and implications \Rightarrow
2. Reduce the scope of all \neg to single term (De Morgan)
3. Make all variable names unique (standardize apart)
4. Move quantifiers left (convert to prenex normal form)
5. Eliminate Existential quantifiers (skolemization)
6. Eliminate Universal quantifiers
7. Convert to conjunction of disjuncts
8. Create separate clause for each conjunct

Converting FOL to CNF: Example 1

Original sentence S:

$$\forall w ([P_1(w) \vee P_2(w) \Rightarrow P_3(w)] \vee [\exists x (\exists y (P_6(x, y) \Rightarrow P_4(w, x)))] \wedge [\forall w (P_5(w))])$$

By De Morgan's Law ($\neg(p \vee q) \equiv \neg p \wedge \neg q$):

$$\forall w ([\neg(P_1(w) \vee P_2(w))] \vee P_3(w)] \vee [\exists x (\exists y (\neg P_6(x, y) \vee P_4(w, x)))] \wedge [\forall w (P_5(w))])$$

becomes:

$$\forall w ([(\neg P_1(w) \wedge \neg P_2(w)) \vee P_3(w)] \vee [\exists x (\exists y (\neg P_6(x, y) \vee P_4(w, x)))] \wedge [\forall w (P_5(w))])$$

Predicate (First-Order) Logic to CNF

Variables and quantifiers are a challenge:

1. Eliminate all equivalences \Leftrightarrow and implications \Rightarrow
2. Reduce the scope of all \neg to single term (De Morgan)
3. **Make all variable names unique (standardize apart)**
4. Move quantifiers left (convert to prenex normal form)
5. Eliminate Existential quantifiers (skolemization)
6. Eliminate Universal quantifiers
7. Convert to conjunction of disjuncts
8. Create separate clause for each conjunct

Converting FOL to CNF: Example 1

Original sentence S:

$$\forall w ([P_1(w) \vee P_2(w) \Rightarrow P_3(w)] \vee [\exists x (\exists y (P_6(x, y) \Rightarrow P_4(w, x)))] \wedge [\forall w (P_5(w))])$$

Variable w (w and w) is bound to two different quantifiers:

$$\forall w ([\neg(P_1(w) \vee P_2(w)) \vee P_3(w)] \vee [\exists x (\exists y (\neg P_6(x, y) \vee P_4(w, x)))] \wedge [\forall w (P_5(w))])$$

Replace w with z and the sentence S becomes:

$$\forall w ([(\neg P_1(w) \wedge \neg P_2(w)) \vee P_3(w)] \vee [\exists x (\exists y (\neg P_6(x, y) \vee P_4(w, x)))] \wedge [\forall z (P_5(z))])$$

Predicate (First-Order) Logic to CNF

Variables and quantifiers are a challenge:

1. Eliminate all equivalences \Leftrightarrow and implications \Rightarrow
2. Reduce the scope of all \neg to single term (De Morgan)
3. Make all variable names unique (standardize apart)
4. **Move quantifiers left (convert to prenex normal form)**
5. Eliminate Existential quantifiers (skolemization)
6. Eliminate Universal quantifiers
7. Convert to conjunction of disjuncts
8. Create separate clause for each conjunct

Converting FOL to CNF: Example 1

Original sentence S:

$$\forall w ([P_1(w) \vee P_2(w) \Rightarrow P_3(w)] \vee [\exists x (\exists y (P_6(x, y) \Rightarrow P_4(w, x)))] \wedge [\forall w (P_5(w))])$$

Quantified variables unique, move quantifiers left (order!):

$$\forall w ([\neg(P_1(w) \vee P_2(w)) \vee P_3(w)] \vee [\exists x (\exists y (\neg P_6(x, y) \vee P_4(w, x)))] \wedge [\forall w (P_5(w))])$$

becomes:

$$\forall w \exists x \exists y \forall z ([(\neg P_1(w) \wedge \neg P_2(w)) \vee P_3(w)] \vee [(\neg P_6(x, y) \vee P_4(w, x))] \wedge [(P_5(z))])$$

Predicate (First-Order) Logic to CNF

Variables and quantifiers are a challenge:

1. Eliminate all equivalences \Leftrightarrow and implications \Rightarrow
2. Reduce the scope of all \neg to single term (De Morgan)
3. Make all variable names unique (standardize apart)
4. Move quantifiers left (convert to prenex normal form)
5. Eliminate Existential quantifiers (skolemization)
6. Eliminate Universal quantifiers
7. Convert to conjunction of disjuncts
8. Create separate clause for each conjunct

Converting FOL to CNF: Example 1

Original sentence S:

$$\forall w ([P_1(w) \vee P_2(w) \Rightarrow P_3(w)] \vee [\exists x (\exists y (P_6(x, y) \Rightarrow P_4(w, x)))] \wedge [\forall w (P_5(w))])$$

We have two existential quantifiers to remove here:

$$\forall w \exists x \exists y \forall z ([(\neg P_1(w) \wedge \neg P_2(w)) \vee P_3(w)] \vee [(\neg P_6(x, y) \vee P_4(w, x))] \wedge [(P_5(z))])$$

and:

$$\forall w \exists x \exists y \forall z ([(\neg P_1(w) \wedge \neg P_2(w)) \vee P_3(w)] \vee [(\neg P_6(x, y) \vee P_4(w, x))] \wedge [(P_5(z))])$$

Both $\exists x$ and $\exists y$ are **inside** the scope of the universal quantifier $\forall w$. We need to use **Skolem function** substitution (Skolemization).

Converting FOL to CNF: Example 1

Original sentence S:

$$\forall w ([P_1(w) \vee P_2(w) \Rightarrow P_3(w)] \vee [\exists x (\exists y (P_6(x, y) \Rightarrow P_4(w, x)))] \wedge [\forall w (P_5(w))])$$

Let's start with $\exists x$ and replace x with a Skolem function:

$$\forall w \exists x \exists y \forall z ([(\neg P_1(w) \wedge \neg P_2(w)) \vee P_3(w)] \vee [(\neg P_6(x, y) \vee P_4(w, x))] \wedge [(P_5(z))])$$

becomes:

$$\forall w \exists y \forall z ([(\neg P_1(w) \wedge \neg P_2(w)) \vee P_3(w)] \vee [(\neg P_6(f(w), y) \vee P_4(w, f(w)))] \wedge [(P_5(z))])$$

Quantified variable x was replaced with Skolem function $f(w)$. Existential quantifier $\exists x$ was removed.

Converting FOL to CNF: Example 1

Original sentence S:

$$\forall w ([P_1(w) \vee P_2(w) \Rightarrow P_3(w)] \vee [\exists x (\exists y (P_6(x, y) \Rightarrow P_4(w, x)))] \wedge [\forall w (P_5(w))])$$

Now: remove $\exists y$ and replace y with a Skolem function:

$$\forall w \exists y \forall z ([(\neg P_1(w) \wedge \neg P_2(w)) \vee P_3(w)] \vee [(\neg P_6(f(w), y) \vee P_4(w, f(w)))] \wedge [(P_5(z))])$$

becomes:

$$\forall w \forall z ([(\neg P_1(w) \wedge \neg P_2(w)) \vee P_3(w)] \vee [(\neg P_6(f(w), g(w)) \vee P_4(w, f(w)))] \wedge [(P_5(z))])$$

Quantified variable y was replaced with Skolem function $g(w)$. Existential quantifier $\exists y$ was removed.

Predicate (First-Order) Logic to CNF

Variables and quantifiers are a challenge:

1. Eliminate all equivalences \Leftrightarrow and implications \Rightarrow
2. Reduce the scope of all \neg to single term (De Morgan)
3. Make all variable names unique (standardize apart)
4. Move quantifiers left (convert to prenex normal form)
5. Eliminate Existential quantifiers (skolemization)
6. **Eliminate Universal quantifiers**
7. Convert to conjunction of disjuncts
8. Create separate clause for each conjunct

Converting FOL to CNF: Example 1

Original sentence S:

$$\forall w ([P_1(w) \vee P_2(w) \Rightarrow P_3(w)] \vee [\exists x (\exists y (P_6(x, y) \Rightarrow P_4(w, x)))] \wedge [\forall w (P_5(w))])$$

Remaining quantified variables are universally quantified:

$$\forall w \forall z ([(\neg P_1(w) \wedge \neg P_2(w)) \vee P_3(w)] \vee [(\neg P_6(f(w), g(w)) \vee P_4(w, f(w)))] \wedge [(P_5(z))])$$

Converting FOL to CNF: Example 1

Original sentence S:

$$\forall w ([P_1(w) \vee P_2(w) \Rightarrow P_3(w)] \vee [\exists x (\exists y (P_6(x, y) \Rightarrow P_4(w, x)))] \wedge [\forall w (P_5(w))])$$

We can simply “drop” universal quantifiers:

$$\forall w \forall z ([(\neg P_1(w) \wedge \neg P_2(w)) \vee P_3(w)] \vee [(\neg P_6(f(w), g(w)) \vee P_4(w, f(w)))] \wedge [(P_5(z))])$$

becomes:

$$([\neg P_1(w) \wedge \neg P_2(w)) \vee P_3(w)] \vee [(\neg P_6(f(w), g(w)) \vee P_4(w, f(w)))] \wedge [(P_5(z))]$$

We are “dropping” universal quantifiers for inferential purposes only. Equivalence is lost, but we can still use the remaining sentence to infer.

Predicate (First-Order) Logic to CNF

Variables and quantifiers are a challenge:

1. Eliminate all equivalences \Leftrightarrow and implications \Rightarrow
2. Reduce the scope of all \neg to single term (De Morgan)
3. Make all variable names unique (standardize apart)
4. Move quantifiers left (convert to prenex normal form)
5. Eliminate Existential quantifiers (skolemization)
6. Eliminate Universal quantifiers
7. Convert to conjunction of disjuncts (CNF)
8. Create separate clause for each conjunct

Converting FOL to CNF: Example 1

Original sentence S:

$$\forall w ([P_1(w) \vee P_2(w) \Rightarrow P_3(w)] \vee [\exists x (\exists y (P_6(x, y) \Rightarrow P_4(w, x)))] \wedge [\forall w (P_5(w))])$$

By Associative Law $((p \vee q) \vee r \Leftrightarrow p \vee (q \vee r))$:

$$([\neg P_1(w) \wedge \neg P_2(w)) \vee P_3(w)] \vee [(\neg P_6(f(w), g(w)) \vee P_4(w, f(w)))] \wedge [P_5(z)]$$

becomes:

$$[P_3(w) \vee (\neg P_1(w) \wedge \neg P_2(w))] \vee [(\neg P_6(f(w), g(w)) \vee P_4(w, f(w)))] \wedge [P_5(z)]$$

Converting FOL to CNF: Example 1

Original sentence S:

$$\forall w ([P_1(w) \vee P_2(w) \Rightarrow P_3(w)] \vee [\exists x (\exists y (P_6(x, y) \Rightarrow P_4(w, x)))] \wedge [\forall w (P_5(w))])$$

By Associative Law $((p \vee q) \vee r \Leftrightarrow p \vee (q \vee r))$:

$$([P_3(w) \vee (\neg P_1(w) \wedge \neg P_2(w))] \vee [(\neg P_6(f(w), g(w)) \vee P_4(w, f(w)))] \wedge [(P_5(z))])$$

becomes:

$$([P_3(w) \vee (\neg P_1(w) \wedge \neg P_2(w))] \vee [(\neg P_6(f(w), g(w)) \vee P_4(w, f(w)))] \wedge [(P_5(z))])$$

Converting FOL to CNF: Example 1

Original sentence S:

$$\forall w ([P_1(w) \vee P_2(w) \Rightarrow P_3(w)] \vee [\exists x (\exists y (P_6(x, y) \Rightarrow P_4(w, x)))] \wedge [\forall w (P_5(w))])$$

By Distributive Law $(p \vee (q \wedge r) \Leftrightarrow (p \vee q) \wedge (p \vee r))$:

$$([P_3(w) \vee (\neg P_1(w) \wedge \neg P_2(w))] \vee [(\neg P_6(f(w), g(w)) \vee P_4(w, f(w)))] \wedge [(P_5(z))])$$

becomes:

$$([(\underline{P_3(w)} \vee \neg \underline{P_1(w)}) \wedge (\underline{P_3(w)} \vee \neg \underline{P_2(w)})] \vee [(\neg P_6(f(w), g(w)) \vee P_4(w, f(w)))] \wedge [(P_5(z))])$$

Converting FOL to CNF: Example 1

Original sentence S:

$$\forall w ([P_1(w) \vee P_2(w) \Rightarrow P_3(w)] \vee [\exists x (\exists y (P_6(x, y) \Rightarrow P_4(w, x)))] \wedge [\forall w (P_5(w))])$$

Let's make some substitutions:

$$([(P_3(w) \vee \neg P_1(w)) \wedge (P_3(w) \vee \neg P_2(w))] \vee [(\neg P_6(f(w), g(w)) \vee P_4(w, f(w)))]) \wedge [(P_5(z))]$$

$$A \equiv (P_3(w) \vee \neg P_1(w))$$

$$B \equiv (P_3(w) \vee \neg P_2(w))$$

$$C \equiv (\neg P_6(f(w), g(w)) \vee P_4(w, f(w)))$$

so the sentence becomes:

$$([A \wedge B] \vee [C]) \wedge [(P_5(z))]$$

Converting FOL to CNF: Example 1

Original sentence S:

$$\forall w ([P_1(w) \vee P_2(w) \Rightarrow P_3(w)] \vee [\exists x (\exists y (P_6(x, y) \Rightarrow P_4(w, x)))] \wedge [\forall w (P_5(w))])$$

By Distributive Law $(p \vee (q \wedge r) \Leftrightarrow (p \vee q) \wedge (p \vee r))$:

$$([A \wedge B] \vee [C]) \wedge [(P_5(z))]$$

becomes:

$$((A \vee C) \wedge (B \vee C)) \wedge [(P_5(z))]$$

where:

$$A \equiv (P_3(w) \vee \neg P_1(w))$$

$$B \equiv (P_3(w) \vee \neg P_2(w))$$

$$C \equiv (\neg P_6(f(w), g(w)) \vee P_4(w, f(w)))$$

Converting FOL to CNF: Example 1

Original sentence S:

$$\forall w ([P_1(w) \vee P_2(w) \Rightarrow P_3(w)] \vee [\exists x (\exists y (P_6(x, y) \Rightarrow P_4(w, x)))] \wedge [\forall w (P_5(w))])$$

Remove substitutions:

$$((A \vee C) \wedge (B \vee C)) \wedge [(P_5(z))]$$

becomes:

$$(((P_3(w) \vee \neg P_1(w)) \vee (\neg P_6(f(w), g(w)) \vee P_4(w, f(w)))) \wedge ((P_3(w) \vee \neg P_2(w)) \vee (\neg P_6(f(w), g(w)) \vee P_4(w, f(w))))) \wedge [(P_5(z))]$$

where:

$$A \equiv (P_3(w) \vee \neg P_1(w))$$

$$B \equiv (P_3(w) \vee \neg P_2(w))$$

$$C \equiv (\neg P_6(f(w), g(w)) \vee P_4(w, f(w)))$$

Converting FOL to CNF: Example 1

Original sentence S:

$$\forall w ([P_1(w) \vee P_2(w) \Rightarrow P_3(w)] \vee [\exists x (\exists y (P_6(x, y) \Rightarrow P_4(w, x)))] \wedge [\forall w (P_5(w))])$$

We can remove some parentheses:

$$(((P_3(w) \vee \neg P_1(w)) \vee (\neg P_6(f(w), g(w)) \vee P_4(w, f(w)))) \wedge ((P_3(w) \vee \neg P_2(w)) \vee (\neg P_6(f(w), g(w)) \vee P_4(w, f(w))))) \wedge [(P_5(z))]$$

becomes:

$$\begin{aligned} & (P_3(w) \vee \neg P_1(w) \vee \neg P_6(f(w), g(w)) \vee P_4(w, f(w))) \\ & \wedge (P_3(w) \vee \neg P_2(w) \vee \neg P_6(f(w), g(w)) \vee P_4(w, f(w))) \\ & \wedge (P_5(z)) \end{aligned}$$

Converting FOL to CNF: Example 1

Original sentence S:

$$\forall w ([P_1(w) \vee P_2(w) \Rightarrow P_3(w)] \vee [\exists x (\exists y (P_6(x, y) \Rightarrow P_4(w, x)))] \wedge [\forall w (P_5(w))])$$

We obtained sentence S in CNF form:

$$\begin{aligned} & (P_3(w) \vee \neg P_1(w) \vee \neg P_6(f(w), g(w)) \vee P_4(w, f(w))) \\ & \wedge (P_3(w) \vee \neg P_2(w) \vee \neg P_6(f(w), g(w)) \vee P_4(w, f(w))) \\ & \wedge (P_5(z)) \end{aligned}$$

Predicate (First-Order) Logic to CNF

Variables and quantifiers are a challenge:

1. Eliminate all equivalences \Leftrightarrow and implications \Rightarrow
2. Reduce the scope of all \neg to single term (De Morgan)
3. Make all variable names unique (standardize apart)
4. Move quantifiers left (convert to prenex normal form)
5. Eliminate Existential quantifiers (skolemization)
6. Eliminate Universal quantifiers
7. Convert to conjunction of disjuncts (CNF)
8. Create separate clause for each conjunct

Converting FOL to CNF: Example 1

Original sentence S:

$$\forall w ([P_1(w) \vee P_2(w) \Rightarrow P_3(w)] \vee [\exists x (\exists y (P_6(x, y) \Rightarrow P_4(w, x)))] \wedge [\forall w (P_5(w))])$$

Let's number all clauses:

$$\begin{aligned} & (P_3(w) \vee \neg P_1(w) \vee \neg P_6(f(w), g(w)) \vee P_4(w, f(w)))_1 \\ & \wedge (P_3(w) \vee \neg P_2(w) \vee \neg P_6(f(w), g(w)) \vee P_4(w, f(w)))_2 \\ & \wedge (P_5(z))_3 \end{aligned}$$

Converting FOL to CNF: Example 2

Original sentence S:

$$\forall x [\forall y (\text{Animal}(y) \Rightarrow \text{Loves}(x, y))] \Rightarrow [\exists y \text{Loves}(y, x)]$$

“Everyone who loves all animals is loved by someone”

Predicate (First-Order) Logic to CNF

Variables and quantifiers are a challenge:

1. **Eliminate all equivalences \Leftrightarrow and implications \Rightarrow**
2. Reduce the scope of all \neg to single term (De Morgan)
3. Make all variable names unique (standardize apart)
4. Move quantifiers left (convert to prenex normal form)
5. Eliminate Existential quantifiers (skolemization)
6. Eliminate Universal quantifiers
7. Convert to conjunction of disjuncts
8. Create separate clause for each conjunct

Converting FOL to CNF: Example 2

Original sentence S:

$$\forall x [\forall y (\text{Animal}(y) \Rightarrow \text{Loves}(x, y))] \Rightarrow [\exists y \text{Loves}(y, x)]$$

By Implication Law ($p \Rightarrow q \equiv \neg p \vee q$):

$$\forall x [\neg \forall y (\text{Animal}(y) \Rightarrow \text{Loves}(x, y))] \vee [\exists y \text{Loves}(y, x)]$$

becomes:

$$\forall x \neg [\forall y (\text{Animal}(y) \Rightarrow \text{Loves}(x, y))] \vee [\exists y \text{Loves}(y, x)]$$

Converting FOL to CNF: Example 2

Original sentence S:

$$\forall x [\forall y (\text{Animal}(y) \Rightarrow \text{Loves}(x, y))] \Rightarrow [\exists y \text{ Loves}(y, x)]$$

By Implication Law ($p \Rightarrow q \equiv \neg p \vee q$):

$$\forall x \neg [\forall y (\text{Animal}(y) \Rightarrow \text{Loves}(x, y))] \vee [\exists y \text{ Loves}(y, x)]$$

becomes:

$$\forall x \neg [(\forall y (\neg \text{Animal}(y) \vee \text{Loves}(x, y)))] \vee [\exists y \text{ Loves}(y, x)]$$

Predicate (First-Order) Logic to CNF

Variables and quantifiers are a challenge:

1. Eliminate all equivalences \Leftrightarrow and implications \Rightarrow
2. Reduce the scope of all \neg to single term (De Morgan)
3. Make all variable names unique (standardize apart)
4. Move quantifiers left (convert to prenex normal form)
5. Eliminate Existential quantifiers (skolemization)
6. Eliminate Universal quantifiers
7. Convert to conjunction of disjuncts
8. Create separate clause for each conjunct

Converting FOL to CNF: Example 2

Original sentence S:

$$\forall x [\forall y (\text{Animal}(y) \Rightarrow \text{Loves}(x, y))] \Rightarrow [\exists y \text{ Loves}(y, x)]$$

By the equivalence $(\neg \forall x (p) \equiv \exists x (\neg p))$:

$$\forall x \neg [(\forall y (\neg \text{Animal}(y) \vee \text{Loves}(x, y)))] \vee [\exists y \text{ Loves}(y, x)]$$

becomes:

$$\forall x [(\exists y \neg (\neg \text{Animal}(y) \vee \text{Loves}(x, y)))] \vee [\exists y \text{ Loves}(y, x)]$$

Converting FOL to CNF: Example 2

Original sentence S:

$$\forall x [\forall y (\text{Animal}(y) \Rightarrow \text{Loves}(x, y))] \Rightarrow [\exists y \text{Loves}(y, x)]$$

By De Morgan's Law ($\neg(p \vee q) \equiv \neg p \wedge \neg q$):

$$\forall x [(\exists y \neg(\neg\text{Animal}(y) \vee \text{Loves}(x, y))) \vee [\exists y \text{Loves}(y, x)]]$$

becomes:

$$\forall x [(\exists y (\neg\neg\text{Animal}(y) \wedge \neg\text{Loves}(x, y))) \vee [\exists y \text{Loves}(y, x)]]$$

Converting FOL to CNF: Example 2

Original sentence S:

$$\forall x [\forall y (\text{Animal}(y) \Rightarrow \text{Loves}(x, y))] \Rightarrow [\exists y \text{ Loves}(y, x)]$$

By Double Negation Law ($\neg(\neg p) \equiv p$):

$$\forall x [(\exists y (\neg\neg\text{Animal}(y) \wedge \neg\text{Loves}(x, y))] \vee [\exists y \text{ Loves}(y, x)]$$

becomes:

$$\forall x [(\exists y (\text{Animal}(y) \wedge \neg\text{Loves}(x, y))] \vee [\exists y \text{ Loves}(y, x)]$$

Predicate (First-Order) Logic to CNF

Variables and quantifiers are a challenge:

1. Eliminate all equivalences \Leftrightarrow and implications \Rightarrow
2. Reduce the scope of all \neg to single term (De Morgan)
3. **Make all variable names unique (standardize apart)**
4. Move quantifiers left (convert to prenex normal form)
5. Eliminate Existential quantifiers (skolemization)
6. Eliminate Universal quantifiers
7. Convert to conjunction of disjuncts
8. Create separate clause for each conjunct

Converting FOL to CNF: Example 2

Original sentence S:

$$\forall x [\forall y (\text{Animal}(y) \Rightarrow \text{Loves}(x, y))] \Rightarrow [\exists y \text{ Loves}(y, x)]$$

Variable y (red y and blue y) is bound to two different quantifiers:

$$\forall x [(\exists y (\neg\neg\text{Animal}(y) \wedge \neg\text{Loves}(x, y))] \vee [\exists y \text{ Loves}(y, x)]$$

Replace blue y with z and the sentence S becomes:

$$\forall x [(\exists y (\text{Animal}(y) \wedge \neg\text{Loves}(x, y))] \vee [\exists z \text{ Loves}(z, x)]$$

Predicate (First-Order) Logic to CNF

Variables and quantifiers are a challenge:

1. Eliminate all equivalences \Leftrightarrow and implications \Rightarrow
2. Reduce the scope of all \neg to single term (De Morgan)
3. Make all variable names unique (standardize apart)
4. **Move quantifiers left (convert to prenex normal form)**
5. Eliminate Existential quantifiers (skolemization)
6. Eliminate Universal quantifiers
7. Convert to conjunction of disjuncts
8. Create separate clause for each conjunct

Converting FOL to CNF: Example 2

Original sentence S:

$$\forall x [\forall y (\text{Animal}(y) \Rightarrow \text{Loves}(x, y))] \Rightarrow [\exists y \text{ Loves}(y, x)]$$

We CAN'T move $\exists z$ left, as it is on the same “level” as $\exists y$:

$$\forall x [(\exists y (\text{Animal}(y) \wedge \neg \text{Loves}(x, y))] \vee [\exists z \text{ Loves}(z, x)]$$

Predicate (First-Order) Logic to CNF

Variables and quantifiers are a challenge:

1. Eliminate all equivalences \Leftrightarrow and implications \Rightarrow
2. Reduce the scope of all \neg to single term (De Morgan)
3. Make all variable names unique (standardize apart)
4. Move quantifiers left (convert to prenex normal form)
5. Eliminate Existential quantifiers (skolemization)
6. Eliminate Universal quantifiers
7. Convert to conjunction of disjuncts
8. Create separate clause for each conjunct

Converting FOL to CNF: Example 2

Original sentence S:

$$\forall x [\forall y (\text{Animal}(y) \Rightarrow \text{Loves}(x, y))] \Rightarrow [\exists y \text{ Loves}(y, x)]$$

We have two existential quantifiers to remove ($\exists y, \exists z$) :

$$\forall x [(\exists y (\text{Animal}(y) \wedge \neg \text{Loves}(x, y))] \vee [\exists z \text{ Loves}(z, x)]$$

and:

$$\forall x [(\exists y (\text{Animal}(y) \wedge \neg \text{Loves}(x, y))] \vee [\exists z \text{ Loves}(z, x)]$$

Both $\exists y$ and $\exists z$ are **inside** the scope of the universal quantifier $\forall x$. We need to use **Skolem function** substitution (Skolemization).

Converting FOL to CNF: Example 2

Original sentence S:

$$\forall x [\forall y (\text{Animal}(y) \Rightarrow \text{Loves}(x, y))] \Rightarrow [\exists y \text{ Loves}(y, x)]$$

Let's start with $\exists y$ and replace y with a Skolem function:

$$\forall x [(\exists y (\text{Animal}(y) \wedge \neg \text{Loves}(x, y))] \vee [\exists z \text{ Loves}(z, x)]$$

becomes:

$$\forall x [(\text{Animal}(F(x)) \wedge \neg \text{Loves}(x, F(x))) \vee [\exists z \text{ Loves}(z, x)]$$

Quantified variable y was replaced with Skolem function $F(x)$. Existential quantifier $\exists y$ was removed.

Converting FOL to CNF: Example 2

Original sentence S:

$$\forall x [\forall y (\text{Animal}(y) \Rightarrow \text{Loves}(x, y))] \Rightarrow [\exists y \text{ Loves}(y, x)]$$

Now, remove $\exists z$ and replace y with a Skolem function:

$$\forall x [(\exists y (\text{Animal}(y) \wedge \neg \text{Loves}(x, y))] \vee [\exists z \text{ Loves}(z, x)]$$

becomes:

$$\forall x [(\text{Animal}(F(x)) \wedge \neg \text{Loves}(x, F(x))) \vee [\text{Loves}(G(x), x)]$$

Quantified variable z was replaced with Skolem function $G(x)$. Existential quantifier $\exists z$ was removed.

Predicate (First-Order) Logic to CNF

Variables and quantifiers are a challenge:

1. Eliminate all equivalences \Leftrightarrow and implications \Rightarrow
2. Reduce the scope of all \neg to single term (De Morgan)
3. Make all variable names unique (standardize apart)
4. Move quantifiers left (convert to prenex normal form)
5. Eliminate Existential quantifiers (skolemization)
6. **Eliminate Universal quantifiers**
7. Convert to conjunction of disjuncts
8. Create separate clause for each conjunct

Converting FOL to CNF: Example 2

Original sentence S:

$$\forall x [\forall y (\text{Animal}(y) \Rightarrow \text{Loves}(x, y))] \Rightarrow [\exists y \text{Loves}(y, x)]$$

Remaining quantified variables are universally quantified:

$$\forall x [(\text{Animal}(F(x)) \wedge \neg \text{Loves}(x, F(x))) \vee [\text{Loves}(G(x), x)]]$$

Converting FOL to CNF: Example 2

Original sentence S:

$$\forall x [\forall y (\text{Animal}(y) \Rightarrow \text{Loves}(x, y))] \Rightarrow [\exists y \text{Loves}(y, x)]$$

We can simply “drop” universal quantifiers:

$$\forall x [(\text{Animal}(F(x)) \wedge \neg \text{Loves}(x, F(x)))] \vee [\text{Loves}(G(x), x)]$$

becomes:

$$[(\text{Animal}(F(x)) \wedge \neg \text{Loves}(x, F(x)))] \vee [\text{Loves}(G(x), x)]$$

We are “dropping” universal quantifiers for inferential purposes only. Equivalence is lost, but we can still use the remaining sentence to infer.

Predicate (First-Order) Logic to CNF

Variables and quantifiers are a challenge:

1. Eliminate all equivalences \Leftrightarrow and implications \Rightarrow
2. Reduce the scope of all \neg to single term (De Morgan)
3. Make all variable names unique (standardize apart)
4. Move quantifiers left (convert to prenex normal form)
5. Eliminate Existential quantifiers (skolemization)
6. Eliminate Universal quantifiers
7. Convert to conjunction of disjuncts
8. Create separate clause for each conjunct

Converting FOL to CNF: Example 2

Original sentence S:

$$\forall x [\forall y (\text{Animal}(y) \Rightarrow \text{Loves}(x, y))] \Rightarrow [\exists y \text{Loves}(y, x)]$$

By Commutative Law ($p \vee q \Leftrightarrow q \vee p$):

$$[(\text{Animal}(F(x)) \wedge \neg \text{Loves}(x, F(x)))] \vee [\text{Loves}(G(x), x)]$$

becomes:

$$[\text{Loves}(G(x), x)] \vee [(\text{Animal}(F(x)) \wedge \neg \text{Loves}(x, F(x)))]$$

Predicate (First-Order) Logic to CNF

Variables and quantifiers are a challenge:

1. Eliminate all equivalences \Leftrightarrow and implications \Rightarrow
2. Reduce the scope of all \neg to single term (De Morgan)
3. Make all variable names unique (standardize apart)
4. Move quantifiers left (convert to prenex normal form)
5. Eliminate Existential quantifiers (skolemization)
6. Eliminate Universal quantifiers
7. Convert to conjunction of disjuncts
8. Create separate clause for each conjunct

Converting FOL to CNF: Example 2

Original sentence S:

$$\forall x [\forall y (\text{Animal}(y) \Rightarrow \text{Loves}(x, y))] \Rightarrow [\exists y \text{Loves}(y, x)]$$

By Distributive Law $(p \vee (q \wedge r) \Leftrightarrow (p \vee q) \wedge (p \vee r))$:

$$[\text{Loves}(G(x), x)] \vee [(\text{Animal}(F(x)) \wedge \neg \text{Loves}(x, F(x)))]$$

becomes:

$$(\text{Loves}(G(x), x) \vee \text{Animal}(F(x))) \wedge (\text{Loves}(G(x), x) \vee \neg \text{Loves}(x, F(x)))$$

Predicate (First-Order) Logic to CNF

Variables and quantifiers are a challenge:

1. Eliminate all equivalences \Leftrightarrow and implications \Rightarrow
2. Reduce the scope of all \neg to single term (De Morgan)
3. Make all variable names unique (standardize apart)
4. Move quantifiers left (convert to prenex normal form)
5. Eliminate Existential quantifiers (skolemization)
6. Eliminate Universal quantifiers
7. Convert to conjunction of disjuncts
8. Create separate clause for each conjunct

Converting FOL to CNF: Example 2

Original sentence S:

$$\forall x [\forall y (\text{Animal}(y) \Rightarrow \text{Loves}(x, y))] \Rightarrow [\exists y \text{Loves}(y, x)]$$

Sentence S is now in CNF form:

$$(\text{Loves}(G(x), x) \vee \text{Animal}(F(x))) \wedge (\text{Loves}(G(x), x) \vee \neg \text{Loves}(x, F(x)))$$

Predicate (First-Order) Logic to CNF

Variables and quantifiers are a challenge:

1. Eliminate all equivalences \Leftrightarrow and implications \Rightarrow
2. Reduce the scope of all \neg to single term (De Morgan)
3. Make all variable names unique (standardize apart)
4. Move quantifiers left (convert to prenex normal form)
5. Eliminate Existential quantifiers (skolemization)
6. Eliminate Universal quantifiers
7. Convert to conjunction of disjuncts
8. Create separate clause for each conjunct

Converting FOL to CNF: Example 2

Original sentence S:

$$\forall x [\forall y (\text{Animal}(y) \Rightarrow \text{Loves}(x, y))] \Rightarrow [\exists y \text{Loves}(y, x)]$$

Let's number all clauses:

$$(\text{Loves}(G(x), x) \vee \text{Animal}(F(x))) \wedge (\text{Loves}(G(x), x) \vee \neg \text{Loves}(x, F(x)))$$

becomes:

$$(\text{Loves}(G(x), x) \vee \text{Animal}(F(x)))_1 \wedge (\text{Loves}(G(x), x) \vee \neg \text{Loves}(x, F(x)))_2$$

Predicate Logic Resolution: Example

Consider following sentences in English

- A. Everyone who loves all animals is loved by someone
- B. Anyone who kills an animal is loved by no one
- C. Jack Loves all animals
- D. Either Jack or Curiosity killed the cat, who is named Tuna

Q. Did Curiosity kill the cat?

FOL: The Resolution Inference Rule

Two clauses, which are assumed to be standardized apart, so **that they share no variables**, can be resolved if they contain complementary literals:

- Propositional literals are complementary if **one is the negation of the other**
- Predicate logic literals are complementary if **one unifies with the negation of the other**

$$(l_1 \vee \dots \vee l_k), (m_1 \vee \dots \vee m_n)$$

$$\text{SUBST}(\theta, l_1 \vee \dots \vee l_{i-1} \vee l_{i+1} \vee \dots \vee l_k \vee m_1 \vee \dots \vee m_{j-1} \vee m_{j+1} \vee \dots \vee m_n)$$

where $\theta = \text{UNIFY}(l_{i-1}, m_j)$.

FOL: The Resolution Inference Rule

For example, the following two clauses:

$[\text{Animal}(\textcolor{red}{F}(\textcolor{red}{x})) \vee \text{Loves}(\textcolor{blue}{G}(\textcolor{blue}{x}), \textcolor{green}{x})]$ and

$[\neg \text{Loves}(\textcolor{blue}{u}, \textcolor{violet}{v}) \vee \neg \text{Kills}(\textcolor{blue}{u}, \textcolor{violet}{v})]$

can be resolved by eliminating complementary literals

$\text{Loves}(\textcolor{blue}{G}(\textcolor{blue}{x}), \textcolor{green}{x})$ and $\neg \text{Loves}(\textcolor{blue}{u}, \textcolor{violet}{v})$

with the unifier

$$\theta = \{\textcolor{blue}{u}/\textcolor{blue}{G}(\textcolor{blue}{x}), \textcolor{violet}{v}/\textcolor{green}{x}\},$$

to produce the resolvent clause:

$[\text{Animal}(\textcolor{red}{F}(\textcolor{red}{x})) \vee \neg \text{Kills}(\textcolor{blue}{G}(\textcolor{blue}{x}), \textcolor{green}{x})]$

Predicate Logic Resolution: Example

Now, let's turn them into predicate logic sentences/KB:

A. $\forall x [\forall y (\text{Animal}(y) \Rightarrow \text{Loves}(x, y))] \Rightarrow [\exists y \text{Loves}(y, x)]$

B. $\forall x [\exists z (\text{Animal}(z) \wedge \text{Kills}(x, z))] \Rightarrow [\forall y \neg \text{Loves}(y, x)]$

C. $\forall x [\text{Animal}(x) \Rightarrow \text{Loves}(\text{Jack}, x)]$

D. $\text{Kills}(\text{Jack}, \text{Tuna}) \vee \text{Kills}(\text{Curiosity}, \text{Tuna})$

E. $\text{Cat}(\text{Tuna})$

F. $\forall x [\text{Cat}(x) \Rightarrow \text{Animal}(x)]$

Q. $\text{Kills}(\text{Curiosity}, \text{Tuna})$, so $\neg Q \equiv \neg \text{Kills}(\text{Curiosity}, \text{Tuna})$

Predicate Logic Resolution: Example

Let's turn them into predicate logic CNF sentences/KB:

A1. $(\text{Animal}(\text{F}(\mathbf{x})) \vee \text{Loves}(\text{G}(\mathbf{x}), \mathbf{x}))$ (A1 and A2 related)

A2. $(\neg \text{Loves}(\mathbf{x}, \text{F}(\mathbf{x})) \vee \text{Loves}(\text{G}(\mathbf{x}), \mathbf{x}))$

B. $(\neg \text{Loves}(\mathbf{y}, \mathbf{x}) \vee \neg \text{Animal}(\mathbf{z}) \vee \neg \text{Kills}(\mathbf{x}, \mathbf{z}))$

C. $(\neg \text{Animal}(\mathbf{x}) \vee \text{Loves}(\text{Jack}, \mathbf{x}))$

D. $(\text{Kills}(\text{Jack}, \text{Tuna}) \vee \text{Kills}(\text{Curiosity}, \text{Tuna}))$

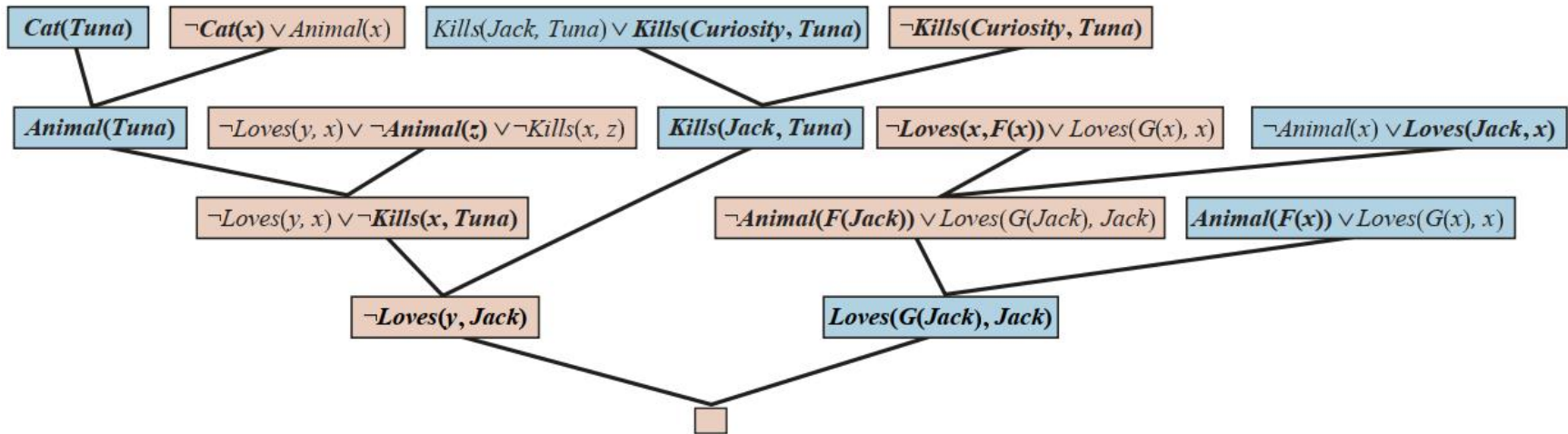
E. $(\text{Cat}(\text{Tuna}))$

F. $(\neg \text{Cat}(\mathbf{x}) \vee \text{Animal}(\mathbf{x}))$

Q. $\text{Kills}(\text{Curiosity}, \text{Tuna})$, so $\neg Q \equiv (\neg \text{Kills}(\text{Curiosity}, \text{Tuna}))$

Predicate Logic Resolution: Example

Resolution process with substitutions:



Notice the use of factoring in derivation of the clause($Loves(G(Jack), Jack)$)