03.09.2026

Normal forms in CPL.

We will describe CNF (conjunctive normal form), DNF (disjunctive normal form), and NNF (negation normal form). To start with, a literal is either an atomic proposition p or its negation $\neg p$.

CNF: A formula C is in CNF if it is a conjunction of clauses, and each clause D is a disjunction of literals:

$$L := p \mid \neg p$$

$$D := L \mid L \vee D$$

$$C := D \mid D \wedge C$$

Examples: (i) $(\neg q \vee p \vee r) \wedge (\neg p \vee r) \wedge q$

(ii) $(p \vee r) \wedge (\neg p \vee r) \wedge (p \vee \neg r)$

Why do we want formulas in CNF?

It is easy to check validity of formulas.

DNF: Can be described dually as disjunctions of conjunctive clauses.

Examples: (i) $(\neg q \wedge p \wedge r) \vee (\neg p \wedge r) \vee q$

NNF: A formula N is in NNF if the negations occur only in front of atomic propositions.

Examples: (i) $(\neg p \wedge q) \wedge (r \rightarrow \neg s)$

Result: Any CPL formula is equivalent to a formula in CNF (DNF / NNF).

H. W. Prove this result.

Resolution

This gives a proof system with no axioms and only one rule.

$$\frac{\alpha \vee \beta \qquad \neg\beta \vee \gamma}{\alpha \vee \gamma} \; \text{Res}$$

This rule is clearly sound.

It turns out that this is all you need, at least to prove that a formula is unsatisfiable: that is the system is used for refutations rather than proofs.

Given α, is α (un)satisfiable?

- Negate the formula to be refuted.
- Convert it to CNF
- Apply the Res rule repeatedly, until either cannot proceed further or, have derived a contradiction.

What do we achieve here?

- If we derive a contradiction, then $\models \alpha$.
- If the procedure gets stuck, we have produced a counter-model for α.
- The procedure is guaranteed to stop.

The main idea

- Each propositional formula α defines a Boolean function $f_\alpha : \{0,1\}^{|P_\alpha|} \to \{0,1\}$, where P_α is the set of all propositional variables in α.

  - If $P_\alpha = \{p_1, \dots, p_k\}$, we can envision f_α as a Boolean table with 2^k rows and $k+1$ columns.

  - To find models of α we only need to consider rows with the last column entry 1.

  - Each such row can be given by a formula $\beta_r = \bigwedge_{i=1}^{k} \gamma_i$, where $\gamma_i = p_i$ if the ith column entry is 1 and $\gamma_i = \neg p_i$, otherwise.

  - We can check that $\models \alpha \leftrightarrow \bigvee_{r:1} \beta_r$. Such a formula is in DNF.

  - We then have a CNF for $\neg\alpha$

Example: $((p \vee q) \wedge (p \to r) \wedge (q \to r)) \to r \;\; (= \alpha)$

We want α to be refuted. We consider $\neg\alpha$.

What is $\neg\alpha$ in CNF?

$\neg\alpha : \;\; (p \vee q) \wedge (\neg p \vee r) \wedge (\neg q \vee r) \wedge \neg r$

$(\neg q \wedge \neg r) \vee (r \wedge \neg r)$

We have 4 clauses : 1. $p \vee q$ 2. $\neg p \vee r$ 3. $\neg q \vee r$ 4. $\neg r$

5. Resolving 1, 2 : we get $q \vee r$

6. Resolving 3, 5 : we get r

7. Resolving 4, 6 : we get $\perp$

Thus, we have $\models \alpha$.

Resolution with quantifiers (FOL without $=$).

- Resolution is similar for FOL, but, what does one do with the variables?
- Suppose we have $\forall x \,(P(x) \to Q(x))$ and $P(c)$.
- We could substitute c for x and get the clauses $\neg P(c) \vee Q(c)$, $P(c)$ and apply propositional resolution to get $Q(c)$.
- The key is to find a proper substitution of variables.
- We can think of substitution as partial maps from variables to terms.
- We will write them as a set of pairs
- Suppose θ, θ' denote substitution
- Given a substitution θ we can lift it to a map on terms (one needs to be careful!)

- Two terms t_1 and t_2 are unifiable if there exists a substitution θ such that $\theta(t_1) = \theta(t_2)$. In this case θ is said to be a unifier of t_1 and t_2.

Examples: 1. $t_1 = x$; $t_2 = y$.

- (x, y) is a unifier,
- (y, x) is also a unifier.
- $\{(x, c), (y, c)\}$ is one more and a simple one.
- $\{(x, f(z)), (y, f(g(c))), (z, g(c))\}$ is a more complicated unifier.

Most general unifier We say θ is a most general unifier (mgu) of terms t_1 and t_2, if for all unifiers θ_1 of t_1 and t_2, there exists a substitution θ_2, such that $\theta_1(t_1) = \theta_2(\theta(t_1))$ and $\theta_1(t_2) = \theta_2(\theta(t_2))$.

Examples: 1. $mgu(f(x), f(c)) = (x, c)$

2. $mgu(h(f(x), y, g(x)), h(f(x), x, g(x))) = (x, y)$ or (y, x).

Thus, mgu is not unique.

3. $mgu(h(g(f(v)), g(u)), h(x, x)) =$

$\{(x, g(f(v))), (u, f(v))\}$

4. $mgu(h(x, f(x)), h(x, x)) = ?$

No mgu exists!

Resolution for FOL.

$$\frac{\alpha \vee \beta_1 \qquad \neg\beta_2 \vee \gamma}{\alpha \vee \gamma} \text{ Res}, \theta, \quad \text{where}$$

θ is an mgu for β_1 and β_2.

Questions:

- What does it mean to say an mgu of literals?
- How do we have β_1 / β_2 as literals?

Note: We already know how to get CNF's.
But how to deal with quantifiers?

Prenex normal form.

An fo-formula φ is said to be in prenex normal form if it is of the form:

$Q_1 x_1 \ldots Q_n x_n \,\varphi'$, where φ' is quantifier-free.

Result: For any fo-formula ϕ, there is a logically equivalent formula in prenex normal form.

The main idea is to use the following logically equivalent formulas.

Q1(a). $\neg \forall x \varphi \leftrightarrow \exists x \neg \varphi$.

Q1(b) $\neg \exists x \varphi \leftrightarrow \forall x \neg \varphi$.

Q2.(a) $(\forall x \varphi \wedge \forall x \psi) \leftrightarrow \forall x (\varphi \wedge \psi)$

Q2.(b) $(\exists x \varphi \vee \exists x \psi) \leftrightarrow \exists x (\varphi \vee \psi)$.

Q3.(a) $(\varphi \to \forall x \psi) \leftrightarrow \forall x (\varphi \to \psi)$

Q3.(b) $(\varphi \to \exists x \psi) \leftrightarrow \exists x (\varphi \to \psi)$

x is not free in φ

Q.4(a) $(\forall x \varphi \to \psi) \leftrightarrow \exists x (\varphi \to \psi)$

Q4.(b) $(\exists x \varphi \to \psi) \leftrightarrow \forall x (\varphi \to \psi)$

x is not free in ψ

H.W. Prove the statement.

Example: $\forall x\big((\exists y\, R(x,y) \wedge \forall y\, \neg S(x,y))$

$\longrightarrow \neg(\exists y\, R(x,y) \wedge P(x,y))\big)$

$\longleftrightarrow \forall x\big((\forall y\, \neg R(x,y) \vee \exists y\, S(x,y))$

$\vee (\forall y\, \neg R(x,y) \vee \neg P(x,y))\big)$

$\longleftrightarrow \forall x(\forall y\, \neg R(x,y) \vee \exists y\, S(x,y)$

$\vee\, \forall y\, \neg R(x,y) \vee \neg P(x,y))$

$\longleftrightarrow \forall x(\forall y\, \neg R(x,y) \vee \exists y\, S(x,y) \vee \neg P(x,y))$

[One more idea: Standardize apart the variables]

$\longleftrightarrow \forall x(\forall y_1 \neg R(x,y_1) \vee \exists y_2 S(x,y_2) \vee \neg P(x,y))$

$\longleftrightarrow \forall x \forall y_1 \exists y_2 (\neg R(x,y_1) \vee S(x,y_2) \vee \neg P(x,y))$

[$Q_1 x\, \varphi \;\underline{op}\; Q_2 y\, \psi \longleftrightarrow Q_1 x\, Q_2 y(\varphi \;\underline{op}\; \psi)$
with $x \neq y$, x not occurring in ψ
and y not occurring in φ]

Skolemization

Consider : $\forall x_1 \cdots \forall x_k \, \exists y \, \varphi$. $(k \geq 0)$

- $k=0$: $\forall x_1 \cdots \forall x_k \, \varphi[c/y]$, c is a new constant symbol.

- $k>0$: Choose a new k-ary function symbol f and write :

$$\forall x_1 \cdots \forall x_k \, \varphi[f(x_1, x_2, \cdots, x_k)/y]$$

- Repeat this process until all $\exists$ are eliminated.

- Drop the universal quantifiers

- Convert to CNF.

Example : 1. $\exists x \forall y \exists z \, R(x, y, z)$

$$\Rightarrow \forall y \exists z \, R(a, y, z)$$

$$\Rightarrow \forall y \, R(a, y, f(y))$$

$$\Rightarrow \{R(a, y, f(y))\}.$$

- Skolemization does not preserve validity.

$\models P(a) \rightarrow \exists x P(x)$.

$\not\models P(a) \rightarrow P(b)$

- Skolemization preserves satisfiability and unsatisfiability. (Check!)

Why does it make sense of talking about mgu's of atomic formulas?

The idea remains the same, only function symbols are replaced by predicate symbols

Example: After PNF and Skolemization procedure, suppose we have the following:

1. $P(u) \vee P(f(u))$

2. $\neg P(v) \vee P(f(w))$

3. $\neg P(x) \vee \neg P(f(x))$

---

4. $P(u) \vee P(f(w))$, Resolving (1), (2) with $v = f(u)$.

5. $P(f(w))$, Factoring 4 with $u = f(w)$

6. $\neg P(f(f(y)))$, Resolving 3 and 5, with $w = y$, $x = f(y)$

7. $\perp$, Resolving 5, 6 with $w = f(y)$

What does factoring mean?

Factoring basically collapses two or more literals into one, by application of mgu.

Remarks: In this discussion we have presented one more proof / refutation system for CPL / FOL which aids in checking (un) satisfiability of formulas. Such methodologies have been applied in SAT solvers among others.

Can we always do this for FOL?

What problems do we face ....