Lecture 12 - FO: Normal forms

Vaishnavi Sundararajan

COL703 - Logic for Computer Science

Recap: Towards a normal form

- Push all quantifiers out into one “block” at the head of the expression
- A **substitution** θ is a partial map from $\mathscr{V}$ to $\mathrm{T}(\Sigma)$, with a finite domain
- $\theta(t) = t$ for a term t in the language, if $\mathsf{vars}(t) \cap \mathsf{dom}(\theta) = \emptyset$
- Often write $t\theta$ to mean $\theta(t)$; $t\theta$ is a “substitution instance” of t
- We often write $\theta = \{t/x \mid x\theta = t \text{ and } x \in \mathsf{dom}(\theta)\}$
- Read $\theta = \{t/x\}$ as “x is replaced by t under θ”
- **Substitution Lemma**: Given an interpretation $\mathscr{I} = ((M, \iota), \sigma)$ for some Σ, a term $t \in \mathrm{T}(\Sigma)$, a formula $\varphi \in \mathsf{FO}_\Sigma$, and a substitution $\{u/x\}$ such that $u^{\mathscr{I}} = m \in M$, the following hold:
  - $(t\{u/x\})^{\mathscr{I}} = t^{\mathscr{I}[x \mapsto m]}$
  - $\mathscr{I} \models \varphi\{u/x\}$ iff $\mathscr{I}[x \mapsto m] \models \varphi$.

Recap: Moving quantifiers out

- Want to move quantifiers into one block at the head of the expression
- **Theorem**: Let $z \notin \mathsf{fv}(\varphi) \cup \mathsf{fv}(\psi) \cup \{x_1, \ldots, x_n\}$, where $n \geqslant 0$. For $Q_i \in \{\forall, \exists\}$ for every $1 \leqslant i \leqslant n$, the following equivalences hold.

$$Q_1 x_1 \ldots Q_n x_n.\ [\neg Q y.\ [\varphi]] \Leftrightarrow Q_1 x_1 \ldots Q_n x_n.\ \overline{Q} y.\ [\neg \varphi]$$

$$Q_1 x_1 \ldots Q_n x_n.\ [\psi \circ Q y.\ [\varphi]] \Leftrightarrow Q_1 x_1 \ldots Q_n x_n.\ Q z.\ [\psi \circ \varphi\{z/y\}]$$

$$Q_1 x_1 \ldots Q_n x_n.\ [Q y.\ [\varphi] * \psi] \Leftrightarrow Q_1 x_1 \ldots Q_n x_n.\ Q z.\ [\varphi\{z/y\} * \psi]$$

$$Q_1 x_1 \ldots Q_n x_n.\ [Q y.\ [\varphi] \supset \psi] \Leftrightarrow Q_1 x_1 \ldots Q_n x_n.\ \overline{Q} z.\ [\varphi\{z/y\} \supset \psi]$$

where $\circ \in \{\wedge, \vee, \supset\}$, and $* \in \{\wedge, \vee\}$, and $\overline{Q} = \begin{cases} \exists & \text{if } Q = \forall \\ \forall & \text{if } Q = \exists \end{cases}$

Prenex Normal Form (PNF)

- PNF: FO expression where all quantifiers "appear at the front"
- $Q_1 x_1 \ldots Q_n x_n. [\varphi]$ is in PNF if φ is **quantifier-free (qf)**.
- Quantifier-free expressions $\subseteq \mathsf{FO}_\Sigma$ generated by the below grammar.

$$\varphi, \psi \coloneqq t_1 \equiv t_2 \,\Big|\, P(t_1, \ldots, t_n) \,\Big|\, \neg\varphi \,\Big|\, \varphi \wedge \psi \,\Big|\, \varphi \vee \psi \,\Big|\, \varphi \supset \psi$$

  where P is an n-ary predicate symbol in Σ, and $t_i \in \mathrm{T}(\Sigma)$ for all $1 \leqslant i \leqslant n$.
- $Q_1 x_1 \ldots Q_n x_n$ is the **prenex**; qf **body** φ contains only equality, predicates, and propositional connectives.
- **Theorem**: For any FO expression φ, there exists a logically equivalent ψ such that ψ is in Prenex Normal Form.

Skolem Normal Form (SNF)

- How does one check for satisfiability of a PNF expression?
- Choice of witness for $\exists$ might depend on value chosen for $\forall$ if $\exists$ appears "deeper" than $\forall$
- Can we reduce (eliminate?!) this sequence of dependencies?
- Recall: For our $\exists y$ example last time, value of m was a function of the value assigned to x. Use this!
- Move to **Skolem Normal Form**

Skolem Normal Form (SNF)

- PNF expression $Q_1 x_1 \dots Q_n x_n. [\varphi]$ is in SNF if $Q_i = \forall$ for every $1 \leqslant i \leqslant n$.
- For $\forall x_1 x_n. [\varphi]$ in SNF, we say that (qf) φ is the **body**
- What are we doing about the existential quantifiers?
- Intuition: Replace every $\exists y$ by a "Skolem function" which computes y using all the (other) variables y depends on.
- Turn $\forall x_1 x_2 \dots x_n . \exists y.\ [\varphi]$ into $\forall x_1 x_2 \dots x_n.\ [\varphi\{f_s(x_1, \dots, x_n)/y\}]$

"For every x there exists y such that $\varphi(x, y)$"

↓

"There is a function sk which maps any x into y_s such that for every x, $\varphi(x, sk(x))$"

Skolem's Theorem

Recall that a model of $\varphi \in \mathsf{FO}_\Sigma$ is an interpretation $\mathcal{I}$ based on a Σ-structure $\mathcal{M}$ such that $\mathcal{I} \vDash \varphi$. We will refer to such a model as being "over Σ".

Theorem: Let $\varphi \in \mathsf{FO}_\Sigma$ be of the form $\forall x_1.\ [\forall x_2.\ [\ldots \forall x_n.\ [\exists y.\ [\psi] \ldots]]]$, such that $x_i \neq x_j$ for any $i \neq j$ and $x_i \neq y$ for any $1 \leqslant i \leqslant n$. Let $\Sigma' = (\mathcal{C}, \mathcal{F} \cup \{sk\}, \mathcal{P})$ where $\Sigma = (\mathcal{C}, \mathcal{F}, \mathcal{P})$, and let $\varphi' = \forall x_1.\ [\forall x_2.\ [\ldots \forall x_n.\ [\psi\{sk(x_1, \ldots, x_n)/y\}] \ldots]] \in \mathsf{FO}_{\Sigma'}$. Then,

1. Every model of φ' over Σ' is a model of φ over Σ'.
2. Every model of φ over Σ can be expanded to a model of φ' over Σ'.

Note that we place no structural restrictions on φ (need not be in any normal form) or on ψ (need not be qf).

Note also that $\mathsf{FO}_\Sigma \subseteq \mathsf{FO}_{\Sigma'}$.

Skolem's Theorem

Proof:

(1) Any interpretation $\mathcal{J}$ over Σ' which satisfies φ' must provide meaning to all of Σ as well (and the extra symbol sk in Σ'). So $\mathcal{J} \models \varphi$ also.

(2) Consider any model $\mathcal{J} = ((M, \iota), \sigma)$ of φ.
$\varphi = \forall x_1.\ [\forall x_2.\ [\ldots \forall x_n.\ [\exists y.\ [\psi] \ldots]]]$, so $\{x_1, \ldots, x_n, y\} \subseteq \mathsf{fv}(\psi)$.
Since $\mathcal{J} \models \varphi$, for every n-tuple $(m_1, \ldots, m_n) \in M^n$, there exists at least one $m_y \in M$ such that $\mathcal{J}[x_1 \mapsto m_1, \ldots, x_n \mapsto m_n, y \mapsto m_y] \models \psi$. Define a function $\mathsf{f} : M^n \to M$ such that f maps every $(m_1, \ldots, m_n)$ to the corresponding m_y.
Define $\iota' = \iota \cup \{sk \mapsto \mathsf{f}\}$. $((M, \iota'), \sigma) \models \varphi'$. ■

Important: φ is satisfiable iff φ' is satisfiable (φ and φ' are **equisatisfiable**)

Skolemization

Theorem: For every sentence $\varphi \in \mathsf{FO}_\Sigma$, there is an algorithm $\mathscr{A}$ to construct an SNF sentence $\varphi_{snf} \in \mathsf{FO}_{\Sigma'}$ such that Σ' contains all the symbols mentioned in Σ, and φ has a model over Σ iff φ_{snf} has a model over Σ'.

Proof:

1. Construct a PNF equivalent ψ_i. If ψ_i does not contain an $\exists$ quantifier, ψ_i is already in SNF. Output ψ_i as φ_{snf}.

2. Otherwise, there is a leftmost existential quantifier such that $\psi_i = \forall x_1. \dots \forall x_n. \exists y. [\xi]$. Skolemize ψ_i to get $\psi_{i+1} = \forall x_1. \dots \forall x_n. [\xi\{sk_i(x_1, \dots, x_n)/y\}]$.

3. ψ_{i+1} has one fewer $\exists$ than ψ_i. Repeat steps 1–3 with ψ_{i+1}.

Example:

$\forall x.\ [\exists y.\ [\forall z.\ [\exists w.\ [P(x, y, z, w)]]]] \rightsquigarrow \forall x.\ [\forall z.\ [P(x, sk_1(x), z, sk_2(x, z))]]$

Use of normal forms

- We wish to establish logical consequence (Given Γ and φ, does $\Gamma \overset{?}{\models} \varphi$?)
- For PL, we did this via CNF and resolution
- Is there an analogue for FO?
- There is a Skolem Normal Form for all FO expressions
- Can cast every FO expression into **Skolem CNF (SCNF)**
- Easy to do; SNF body has a CNF equivalent
- Can we perform resolution on an SCNF expression?
- Need to handle quantifiers and free variables.

Imagine there's no variables...

- Consider the set $\mathrm{T}^g(\Sigma)$ of all **ground** terms (i.e. without variables) over $\Sigma = (\mathscr{C}, \mathscr{F}, \mathscr{P})$ where $\mathscr{C} \neq \emptyset$.
- Expressions of the following forms are called **ground literals**
  - $P(t_1, \ldots, t_n)$ and $\neg P(t_1, \ldots, t_n)$, where $P \in \mathscr{P}$ and $t_1, \ldots, t_n \in \mathrm{T}^g(\Sigma)$
- $\mathrm{T}^g(\Sigma)$ generated by the grammar $t_1, \ldots, t_n \coloneqq c \in \mathscr{C} \mid f(t_1, \ldots, t_n)$
- $\mathrm{T}^g(\Sigma)$ is called the **Herbrand universe** of FO_Σ
- A **Herbrand structure** is $(\mathrm{T}^g(\Sigma), \iota_H)$ where ι_H gives meaning to the constant and function symbols in Σ as follows.
  - $\iota_H(c) = c$, for every $c \in \mathscr{C}$
  - $\iota_H(f) = f$, for every $f \in \mathscr{F}$
- Can add similar meaning to symbols in $\mathscr{P}$, and get a **Herbrand base**.
- Ignore $\equiv$ for the moment; we will handle it later.

Herbrand interpretation

- An assignment with a Herbrand base will yield an interpretation
- We interpreted c to be c and f to be f itself, under ι_H
- So what should our assignment function be a map (from $\mathscr{V}$) to?

Herbrand interpretation

- An assignment with a Herbrand base will yield an interpretation
- We interpreted c to be c and f to be f itself, under ι_H
- So what should our assignment function be a map (from $\mathscr{V}$) to?
- A **Herbrand interpretation** over Σ is of the form $((\mathrm{T}^g(\Sigma), \iota_H), \sigma_H)$, where $\sigma_H : \mathscr{V} \to \mathrm{T}^g(\Sigma)$.
- A **Herbrand model** for $\varphi \in \mathsf{FO}_\Sigma$ is a Herbrand base (which assigns meaning to symbols in $\mathscr{P}$) along with σ_H such that φ is made true.
- Can lift this to sets of expressions as usual.

Pourquoi, Herbrand?

To talk about the satisfiability and validity of sets of ground qf formulae.

Theorem: Let $\Sigma = (\mathscr{C}, \mathscr{F}, \mathscr{P})$ where $\mathscr{C} \neq \emptyset$, and let $L = \{\ell_1, \dots, \ell_n\}$ be a non-empty finite set of ground literals. Then,

1. $\bigwedge\limits_{1 \leqslant i \leqslant n} \ell_i$ is satisfiable iff L does not contain both a literal and its negation.
2. $\bigwedge\limits_{1 \leqslant i \leqslant n} \ell_i$ is never valid.
3. $\bigvee\limits_{1 \leqslant i \leqslant n} \ell_i$ is always satisfiable.
4. $\bigvee\limits_{1 \leqslant i \leqslant n} \ell_i$ is valid iff L contains both a literal and its negation.

Models for ground qf formulae

Proof sketch: We only show one case here. The others are easy and left as an **exercise**. Note that the literals in L are **ground**.

$(1, \Leftarrow)$: Suppose L does not contain $\{\ell, \neg\ell\}$ for any literal ℓ. We define a Herbrand model H for L as follows.

Start with $(\mathrm{T}^g(\Sigma), \iota_H)$, and construct a Herbrand base by assigning meaning to symbols in $\mathscr{P}$. Let $P \in \mathscr{P}$ be an m-ary predicate symbol. Define

$$\mathsf{P}_\mathsf{H} = \{(t_1, \dots, t_m) \in (\mathrm{T}^g(\Sigma))^m \mid P(t_1, \dots, t_m) \in L\}$$

If $p(t_1, \dots, t_m) \in L$, then $(t_1, \dots, t_m) \in \mathsf{P}_\mathsf{H}$ and $H \vDash p(t_1, \dots, t_m)$. However, if $\neg p(t_1, \dots, t_m) \in L$, then $p(t_1, \dots, t_m) \notin L$ (since L does not contain a literal and its negation), and so $(t_1, \dots, t_m) \notin \mathsf{P}_\mathsf{H}$ and $H \nvDash p(t_1, \dots, t_m)$. So $H \vDash \bigwedge_{1 \leqslant i \leqslant n} \ell_i$.

Herbrand's Theorem

Theorem: Let $\Sigma = (\mathscr{C}, \mathscr{F}, \mathscr{P})$ where $\mathscr{C} \neq \emptyset$. Let $\varphi = \forall x_1 \dots x_n.\ [\psi] \in \mathsf{FO}_\Sigma$ be a sentence in SNF. Then, the following are equivalent.

1. φ has a model $\longrightarrow$ 3. Γ^g has a model

   $\uparrow$ $\uparrow$

2. φ has a Herbrand model $\longrightarrow$ 4. Γ^g has a Herbrand model

where $\Gamma^g = \{\psi\{t_1/x_1, \dots, t_n/x_n\} \mid \{t_1, \dots, t_n\} \subseteq \mathrm{T}^g(\Sigma)\}$.

Proof strategy: (2) implies (1) and (4) implies (3).

If φ has a Herbrand model, ψ is made true under all possible assignments of x_i to some term $u_i \in \mathrm{T}^g(\Sigma)$. In particular, ψ is made true under the assignment which maps x_i to t_i for each i, so ψ has a Herbrand model, and so does any expression in Γ^g, by the Substitution Lemma. So (2) implies (4).

Similarly, (1) implies (3).

So to prove the equivalence of (1)–(4), it is enough to show that (3) implies (2).

Herbrand's Theorem

Proof of Herbrand's Theorem: We want to show that if Γ^g has a model, then φ has a Herbrand model.
Let $\mathcal{I} \models \Gamma^g$. We start with $(\mathrm{T}^g(\Sigma), \iota_H)$ and construct a Herbrand base by assigning meaning to symbols in $\mathscr{P}$. Let $P \in \mathscr{P}$ be an m-ary predicate symbol. Define $\mathsf{P}_\mathsf{H} = \{(t_1, \dots, t_m) \in (\mathrm{T}^g(\Sigma))^m \mid \mathcal{I} \models P(t_1, \dots, t_m)\}$.

There are no free variables in φ, so this Herbrand base (along with any assignment σ_H) satisfies all the **atomic sentences** satisfied by $\mathcal{I}$.

Exercise: Lift by induction to **arbitrary** sentences in SNF.

Thus, φ has a Herbrand model.

Using Herbrand's Theorem

- A sentence $\varphi \in \mathsf{FO}_\Sigma$ is satisfiable iff its SNF form φ_{snf} is satisfiable iff Γ^g is satisfied by a Herbrand model.
- Γ^g is an infinite set of ground qf expressions, if there is even one function symbol in $\mathcal{F}$
- How do we check for satisfiability of Γ^g?

Using Herbrand's Theorem

- A sentence $\varphi \in FO_\Sigma$ is satisfiable iff its SNF form φ_{snf} is satisfiable iff Γ^g is satisfied by a Herbrand model.
- Γ^g is an infinite set of ground qf expressions, if there is even one function symbol in $\mathcal{F}$
- How do we check for satisfiability of Γ^g?
- What do we know about the satisfiability of an infinite set of ground qf expressions?

Using Herbrand's Theorem

- A sentence $\varphi \in \mathsf{FO}_\Sigma$ is satisfiable iff its SNF form φ_{snf} is satisfiable iff Γ^g is satisfied by a Herbrand model.
- Γ^g is an infinite set of ground qf expressions, if there is even one function symbol in $\mathcal{F}$
- How do we check for satisfiability of Γ^g?
- What do we know about the satisfiability of an infinite set of ground qf expressions?
- Use Compactness Theorem, in the contrapositive.
- Check all finite subsets to see if any unsatisfiable.
- Use resolution!