First-Order Logic

First-Order Logic

$\forall person\ \text{ItIsRaining}() \rightarrow \text{IsWet}(person)$

- Things:
  - constants: *Fred*, *Car54*
  - functions (thing → thing): $\text{MotherOf}(Fred)$, $\text{NextTo}(Fred, House)$
- Relations:
  - predicates (thing → `bool`): $\text{IsWet}(Fred)$, $\text{LivesIn}(\text{MotherOf}(Fred), House)$
- Complex expressions:
  - connectives: $\text{IsWet}(Fred) \vee \text{OnRoad}(Car54)$
  - variables: $\exists person$
  - quantifiers: $\forall person\ \text{IsWet}(person)$

FOL Examples

$$\forall person \forall time$$
$$(\text{IsRaining}(time) \land \neg \exists umbrella\ \text{Holding}(person, umbrella, time)) \rightarrow$$
$$\text{IsWet}(person, time)$$

Fred loves Mary.
All crows are black.
Whales are mammals that live in the water.
Mary likes the color of one of Fred's ties.
I can't hold more than one thing at a time.

Power of FOL

- Indirect knowledge: $\text{Tall}(\mathit{MotherOf}(\mathit{Fred}))$
- Counterfactuals: $\neg\text{Tall}(\mathit{John})$
- Partial knowledge: $\text{IsSisterOf}(b,a) \vee \text{IsSisterOf}(c,a)$
- Partial knowledge: $\exists x\ \text{IsSisterOf}(x,a)$

Reasoning in FOL

Clauses

1. Eliminate $\leftrightarrow$: convert $a \leftrightarrow b$ to $a \rightarrow b \wedge b \rightarrow a$
2. Eliminate $\rightarrow$: convert $a \rightarrow b$ to $\neg a \vee b$
3. Move $\neg$ inwards: eliminate double negatives and apply De Morgan's law
4. Standardize variables – rename so variables are unique
5. Skolemize existential variables: convert $\exists x\ \text{Rich}(x)$ to $\text{Rich}(y)$, where y is a newly named constant
6. Drop universal qualifiers: convert $\forall x\ \text{Person}(x)$ to $\text{Person}(x)$
7. Distribute $\vee$: convert $(a \wedge b) \vee c$ to $(a \vee c) \wedge (b \vee c)$

Example

1. Cows like grass.
2. Cows eat everything they like.
3. Martin is a cow.

Prove that Martin eats grass.

Resolving the truth of a query

RESOLUTION(KB, α)

$clauses \leftarrow$ set of clauses in $KB \wedge \neg\alpha$
$new \leftarrow \{\}$
while *true* **do**
    for each pair of clauses C_i, C_j in *clauses* **do**
        $resolvents \leftarrow$ RESOLVE(C_i, C_j)
        if $\square \in resolvents$ **then**
            return true
        $new \leftarrow new \cup resolvents$
    if $new \subseteq clauses$ **then**
        return false
    $clauses \leftarrow clauses \cup new$

KB is the knowledge base, α is the query

Unifying Two Terms

UNIFY$(t1, t2)$

```
if One term is a constant then
    if Other term is a constant then
        return t1 = t2
    if Other term is a function then
        return fail
    if Other term is a variable then
        substitute constant for variable
else if One term is a function then
    if Other term is a function then
        return UNIFY(arg lists)
    if Other term is a variable then
        if variable occurs in function then
            return fail
        else substitute function for variable
else
    substitute one variable for the other
```

Example

1. Anyone who reads is literate.
2. Whales are not literate.
3. Some whales are intelligent.
4. Prove: someone intelligent cannot read.

Conversion to CNF

 Frank Kreimendahl | *kreimendahlf@wit.edu*

Basis for Refutation

Remember $KB \models \alpha$ iff $KB \rightarrow \alpha$ is valid

1. Assume $KB \models \alpha$
2. If a model i satisfies KB, then i satisfies α
3. If i satisfies α, then it doesn't satisfy $\neg\alpha$
4. Therefore no model satisfies KB and $\neg\alpha$
5. Therefore $KB \wedge \neg\alpha$ is unsatisfiable
6. Therefore we can derive $\Box$ from $KB \wedge \neg\alpha$