32002: AI (FOPL, Unification, Bound and Free variables) **Spring 2014**

Lecture 8: January 16, 2014

Lecturer: K.R. Chowdhary *: Professor of CS (GF)*

Disclaimer: *These notes have not been subjected to the usual scrutiny reserved for formal publications. They may be distributed outside this class only with the permission of the Instructor.*

8.1 Bound and Free Variables

A variable in a wff is bound if it is within the scope of a quantifier naming the variable, otherwise the variable is free. For example, in $\forall_x(p(x) \rightarrow q(x, y))$, x is bound and y is free; in $\forall_x(p(x) \rightarrow q(x)) \rightarrow r(x)$, the x in $r(x)$ is free variable. In the latter case it is better to rename the variable to remove the ambiguity, and we rephrase this statement as $\forall_x(p(x) \rightarrow q(x)) \rightarrow r(z)$. An expression can be evaluated only when all the variables in that are bound.

8.2 Unification

In order to apply the rules of inference, an inference system must be able to determine when two expressions match. Unification is an algorithm for determining the substitutions needed to make two predicate calculus expressions match.

Generally, in a problem solving process there are multiple inferences and, which causes multiple recursive unifications. Once a value is assigned to a variable, the variable is called *bound*. When it is bound, further unification and inferences must take the value of the binding into account.

Following are some valid substitution instances of $foo(X, a, zoo(Y))$:

1. $foo(fred, a, zoo(Z))$, where $fred$ is substituted for X and Z for Y, i.e., $\{fred/X, Z/Y\}$
2. $foo(W, a, zoo(jack))$, where $\{W/X, jack/Y\}$
3. $foo(Z, a, zoo(moo(Z)))$, where $\{Z/X, moo(Z)/Y\}$

We use the notation X/Y to indicate that X is substituted for the variable Y. We also refer to these as bindings, so Y is bound to X. A variable cannot be unified with a term containing that variable. So X cannot be substituted for $p(X)$, because this would create an infinite regression: $p(p(p(...X)...))$.

If S and S' are two substitution sets, then the composition of S and $S^{'}$, i.e., $SS^{'}$, is obtained by applying S to the elements of $S^{'}$ and adding the result to S. For example, the three sets $\{X/Y, W/Z\}$, $\{V/X\}$, $\{a/V, f(b)/W\}$ are together equivalent to $\{a/Y, f(b)/Z\}$.

If S is any unifier of a set of expressions E, and G is the most general unifier (mgu) of that set of expressions, then for S applied to E there exists another unifier $S^{'}$ such that $ES = EGS^{'}$ (where $GS^{'}$ is the composition of unifiers). In simple words, an mgu is just what its name implies: the most general set of substitutions that can be found for the two expressions.

Example 8.1 *Given unifier, obtain a* **more** *general unifier.*

Suppose you have two expressions $p(X)$ an $p(Y)$. One way to unify these is to substitute any constant expression for X and Y: $\{fred/X,\ fred/Y\}$. But this is not the most general unifier, because if we substitute any variable for X and Y, we get a more general unifier: $\{Z/X, Z/Y\}$. The first unifier is a valid unifier, but it would lessen the generality of inferences that we might want to make.

Let $E = \{p(X), p(Y)\}$

Let $S = \{fred/X, fred/Y\}$

Let $G = \{Z/X, Z/Y\}$

Now, let $S^{'} = \{fred/Z\}$

Then $ES = \{p(fred), p(fred)\}$

and $GS^{'} = \{fred/X, fred/Y\}$

and therefore $EGS^{'} = \{p(fred), p(fred)\} = ES$

So, given a unifier, you can always create a more general unifier by means of composition. □

References

[1] Chowdhary K.R. (2020) Logic and Reasoning Patterns. In: Fundamentals of Artificial Intelligence. Springer, New Delhi. https://doi.org/10.1007/978-81-322-3972-7_3