CS-208:Artificial Intelligence
Topic-13 Computational Predicate

Computational Predicates

When the number of facts are not very large and these facts are sufficiently unstructured among themselves then we can express as a combination of individual predicates.

But, suppose we want to represent simple facts such as

gt(1,0)←*1 is greater than 0*	lt(0,1) ← *0 is less than 1*
gt(2,0)	lt(0,2)
gt(2,1)	lt(1,2)
etc.	

- ✗ The above types of facts are infinitely many of them.
- ✗ It would be extremely inefficient to store explicitly as large set of statements.
- ✓ Easy way is to compute each one of these facts as we need it and express them as computational predicates(these computational predicates invoke a procedure instead of searching in the database).

Example for usage of Computational Predicates

1. Marcus was a man
2. Marcus was a Pompeian
3. Marcus was born in 40 AD
4. All men are mortal.
5. All Pompeian died when the volcano erupted in 79AD
6. No mortal lives longer than 150 years.
7. It is now 2020

Is Marcus alive ?

Converting in to predicate statements

1. Man(Marcus)
2. Pompeian(Marcus)
3. born(Marcus, 40)
4. $\forall x$: Man(x) $\rightarrow$ mortal(x)
5. erupted(volcano, 79) $\wedge$ $\forall x$: [Pompeian(x) $\rightarrow$ died(x, 79)]

   5a. erupted(volcano, 79) 5b. $\forall x$: Pompeian(x) $\rightarrow$ died(x, 79)
6. $\forall x$: $\forall t_1$: $\forall t_2$: mortal(x)$\wedge$ born(x, t_1) $\wedge$ gt(t_2-t_1, 150) $\rightarrow$ dead(x, t_2)
7. now = 2020
8. $\forall x$: $\forall t_1$: $\forall t_2$: died(x, t_1) $\wedge$ gt(t_2, t_1) $\rightarrow$ dead(x, t_2)
9. $\forall x$: $\forall t$: alive(x, t) $\rightarrow$ $\neg$dead(x, t)

Formal Proof -1

(by reasoning backward from the desired goal)

alive(Marcus)

↓ (9)

dead(Marcus, now)

↓(8)

died(Marcus, t_1) ∧ gt(now, t_1)

↓(5b)

Pompeian(Marcus) ∧ gt(now, 79)

↓(2)

gt(now, 79)

↓(7)

gt(2020, 79)

↓

Nil

In order to complete the proof we need to add two more facts

8. If some on dies, then he is dead at all later time

9. Alive means not dead

After adding this fact, the last goal has been satisfied and able to produce the proof that Marcus is not alive now

The term Nil at the end indicates that the proof has succeeded.

The sentence *All Pompeian died when the volcano erupted in 79AD* clearly asserts two facts represented below :

erupted(volcano, 79)

∀x: Pompeian(x) → died(x, 79)

Formal Proof -2

(by reasoning backward from the desired goal)

¬ alive(Marcus)

↓ (9)

dead(Marcus, now)

↓(6)

mortal(Marcus)∧ born(Marcus, t_1) ∧ gt(now-t_1, 150)

↓(4)

Man(Marcus)∧ born(Marcus, t_1) ∧ gt(now-t_1, 150)

↓(1)

born(Marcus, t_1) ∧ gt(now-t_1, 150)

↓(3)

gt(now-40, 150)

↓(7)

Gt(2020-40, 150)

↓

Gt (1980, 150)

↓

Nil

$a \wedge b \rightarrow c$

Here a and b are independent sub-goals if they do not share the same bound variables

Even very simple conclusions require many steps to prove,

A variety of process (such as Matching, Substitutions and application of modus ponens/modus tolens) are involved in the production of proof.