Family tree implemented in Prolog

I.Abstract

***Implement and define family relationships by using Prolog clauses.***

The system uses a set of atoms (e.g. Man, Female and Parent) which are enough for fully defining complex relations for a family (e.g. Aunt, Ancestor etc.). Provided the correct input and existing knowledge base the system can test different relations between members of a family and return an Boolean answer.

II.Syntax

The project is composed of atoms and rules that combine multiple atoms in order to form new complex relations. This way, the set of rules can be divided into two groups, one being independent and the other one dependent.

The clauses provided in the knowledge base can be classified as single input and double input ones:

a)Single input ones(e.g. man(X) ) take as input a name and return an information that is valid only for the person associated with that name (if the man clause is true we know for sure that X is a man but we don’t know anything else about the others)

b)Double input clauses(e.g. sibling(X,Y) ) take as input two names and return True if the states relation hold between the two persons (in this example we ask if X is the sibling of Y).

III.Result section

1.Atoms-represent the core of the knowledge base, being independent of other clauses. Their only purpose is to store data, serving as a base for more complex clauses.

This project includes 3 different atoms, each storing a different information about a family member

1. Man clause-returns True if the provided name is associated to a man.

e.g. man("Ahmed").

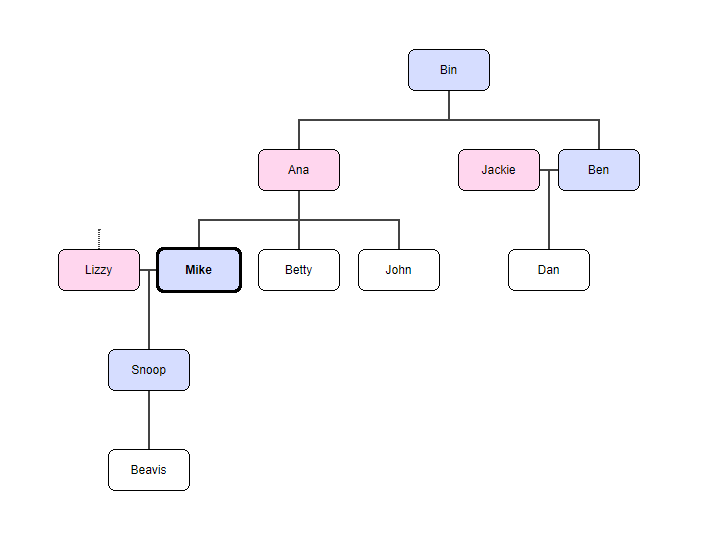
b) Female clause-returns True if the provided name is associated to a woman

e.g. female("Amal").

c) Parent clause-accepts as an input two names and returns true if the person associated with the name is the parent of the second person

e.g parent(“Ben”,”Dan”). -> this states that Ben is the parent of Dan

2.Complex relations combine different atoms in order to form new relations



Considering the provided family tree we can derive and check different relations between members of the family

e.g.

1. Ask who is Ana’s father

father(X,”Ana”).

->returns X=“Bin”

2.Ask who is John’s cousin

Cousin(“John”,X).

->returns X=“Dan”

3. Find all ancestors of a person, in this example Beavis

allAncestors(“Beavis”,R)

->returns R = ["Snoop", "Mike", "Lizzy", "Ana", "Bin"].

4.Check if two persons are relative

relatedCheck(“Ben”,”Lizzy”)

->returns False

All queries presented in the KB are working properly, but I wanted to develop further the app and design a clause that finds all the relatives of a person but I need to modify the checkRelative clause a bit because the result are not good.

IV.Discussion

Pros:

1.Heavily independent KB-it depends only on the three atoms in order to work so adding a new person or relation is easy.

2.Low coupling of clauses, I tried to maximize the use of the basic atoms whilst not depending on other clauses.

3.Easy to formulate new queries because every query is modular.

Cons:

1.Gets cluttered with the addition of new clauses

2.Can’t add a person by directly using a complex clause, instead it needs to be added by atoms.(An inference mechanism would be good for this in order to discover new relations automatically )

V.Apendix

An appendix with the KB code can be found in this folder with the name “family.pl”.