Lab2

Name: Zeng Yuhang

ID: 222320008

Problem statement

The main purpose of this lab work is to learn how to use 'Semantic' program and to understand the differences between prolog and semantic. We should first transfer the knowledge base in prolog to the 'Semantic' environment, and then add attributes with some rules to objects. The first step aims to let us know the syntax differences between them, and the second step shows some new capabilities of semantic network.

Knowledge base text

############################ FamilyTree.pl ##############################
t("family_tree").
onto("Rules.pl").

f("yuhang","has_sex","male").
f("youlin","has_sex","male").
f("youquan","has_sex","male").
f("tao","has_sex","male").
f("qingming","has_sex","male").
f("wei","has_sex","male").
f("du","has_sex","male").
f("heng","has_sex","male").
f("shengqiang","has_sex","male").
f("hang","has_sex","male").
f("yueyang","has_sex","male").
f("anran","has_sex","male").
f("jiayan","has_sex","male").
f("tinggao","has_sex","male").
f("dayou","has_sex","male").

f("yuxi","has_sex","female").
f("li","has_sex","female").
f("yesha","has_sex","female").
f("jun","has_sex","female").
f("xiujuan","has_sex","female").
f("jiafeng","has_sex","female").
f("yuehong","has_sex","female").
f("youping","has_sex","female").
f("lu","has_sex","female").
f("yi","has_sex","female").
f("yue","has_sex","female").
f("xiaoou","has_sex","female").
f("shuang","has_sex","female").
f("daobi","has_sex","female").

f("longhui","has_sex","female").

f("tinggao","is_father_of","youquan").
f("tinggao","is_father_of","youping").
f("tinggao","is_father_of","youlin").
f("youquan","is_father_of","yue").
f("tao","is_father_of","du").
f("youlin","is_father_of","yuhang").
f("youlin","is_father_of","yuxi").
f("dayou","is_father_of","jun").
f("dayou","is_father_of","li").
f("dayou","is_father_of","wei").
f("qingming","is_father_of","heng").
f("qingming","is_father_of","lu").
f("wei","is_father_of","yi").
f("wei","is_father_of","yueyang").
f("shengqiang","is_father_of","anran").
f("hang","is_father_of","jiayan").

f("longhui","is_mother_of","youquan").
f("longhui","is_mother_of","youping").
f("longhui","is_mother_of","youlin").
f("yuehong","is_mother_of","yue").
f("youping","is_mother_of","du").
f("youping","is_mother_of","shuang").
f("li","is_mother_of","yuhang").
f("li","is_mother_of","yuxi").
f("daobi","is_mother_of","jun").
f("daobi","is_mother_of","li").
f("daobi","is_mother_of","wei").
f("jun","is_mother_of","heng").
f("jun","is_mother_of","lu").
f("xiujuan","is_mother_of","yi").
f("xiujuan","is_mother_of","yueyang").
f("lu","is_mother_of","anran").
f("yi","is_mother_of","jiayan").

f("jiafeng","is_spouse_of","wei").
f("xiaoou","is_spouse_of","du").

f("yesha","is_lover_of","yuhang").

f("yuhang","is_a","person").
f("yesha","is_a","person").

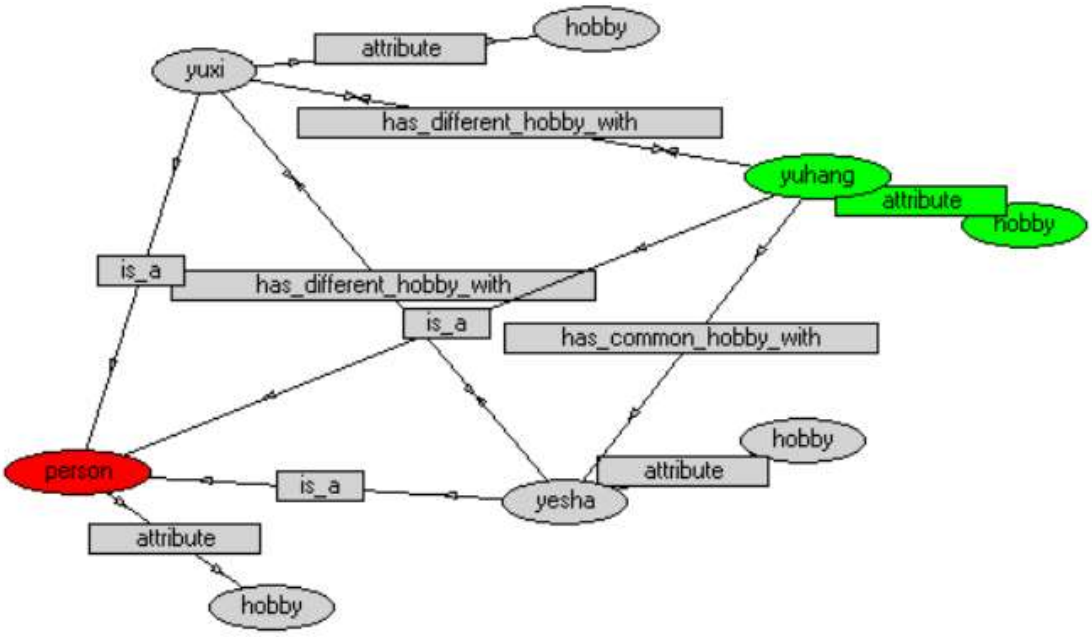
```
f("yuxi","is_a","person").

f("yuhang:hobby","is","sports").
f("yesha:hobby","is","sports").
f("yuxi:hobby","is","painting").
############################### Rules.pl ###############################
r([t("?x","is_father_of","?y")],[t("?x","is_parent_of","?y")]).
r([t("?x","is_mother_of","?y")],[t("?x","is_parent_of","?y")]).
r([t("?x","is_father_of","?y")],[t("?y","is_child_of","?x")]).
r([t("?x","is_mother_of","?y")],[t("?y","is_child_of","?x")]).
r([t("?x","is_child_of","?y"),t("?x","has_sex","male")],[t("?x","is_son_of","?y")]).
r([t("?x","is_child_of","?y"),t("?x","has_sex","female")],[t("?x","is_daughter_of","?y")]).
r([t("?x","is_father_of","?y"),t("?y","is_parent_of","?z")],[t("?x","is_grandfather_of","?z")]).
r([t("?x","is_mother_of","?y"),t("?y","is_parent_of","?z")],[t("?x","is_grandmother_of","?z")]).
r([t("?x","is_child_of","?y"),t("?z","is_child_of","?y"),t("?x","differs","?z")],[t("?x","is_sibling_of","?z"
)]).
r([t("?x","is_parent_of","?y"),t("?z","is_sibling_of","?x"),t("?z","has_sex","male")],[t("?z","is_uncle_
of","?y")]).
r([t("?x","is_parent_of","?y"),t("?z","is_sibling_of","?x"),t("?z","has_sex","female")],[t("?z","is_aunt
_of","?y")]).
r([t("?x","is_parent_of","?y"),t("?z","is_parent_of","?w"),t("?x","is_sibling_of","?z")],[t("?y","is_cous
in_of","?w")]).
r([t("?x","is_parent_of","?y"),t("?z","is_parent_of","?y"),t("?x","differs","?z")],[t("?x","is_spouse_of"
,"?z")]).

c("person").

o("person","attribute","hobby").

r([t("?a","is_a","?b"), t("?b","attribute","?c")], [t("?a","attribute","?a:?c")]).
r([t("?a:hobby","is","?b"),t("?c:hobby","is","?b"),t("?a","differs","?c")],[t("?a","has_common_hobb
y_with","?c")]).
r([t("?a:hobby","is","?b"),t("?c:hobby","is","?d"),t("?b","differs","?d")],[t("?a","has_different_hobb
y_with","?c")]).
```

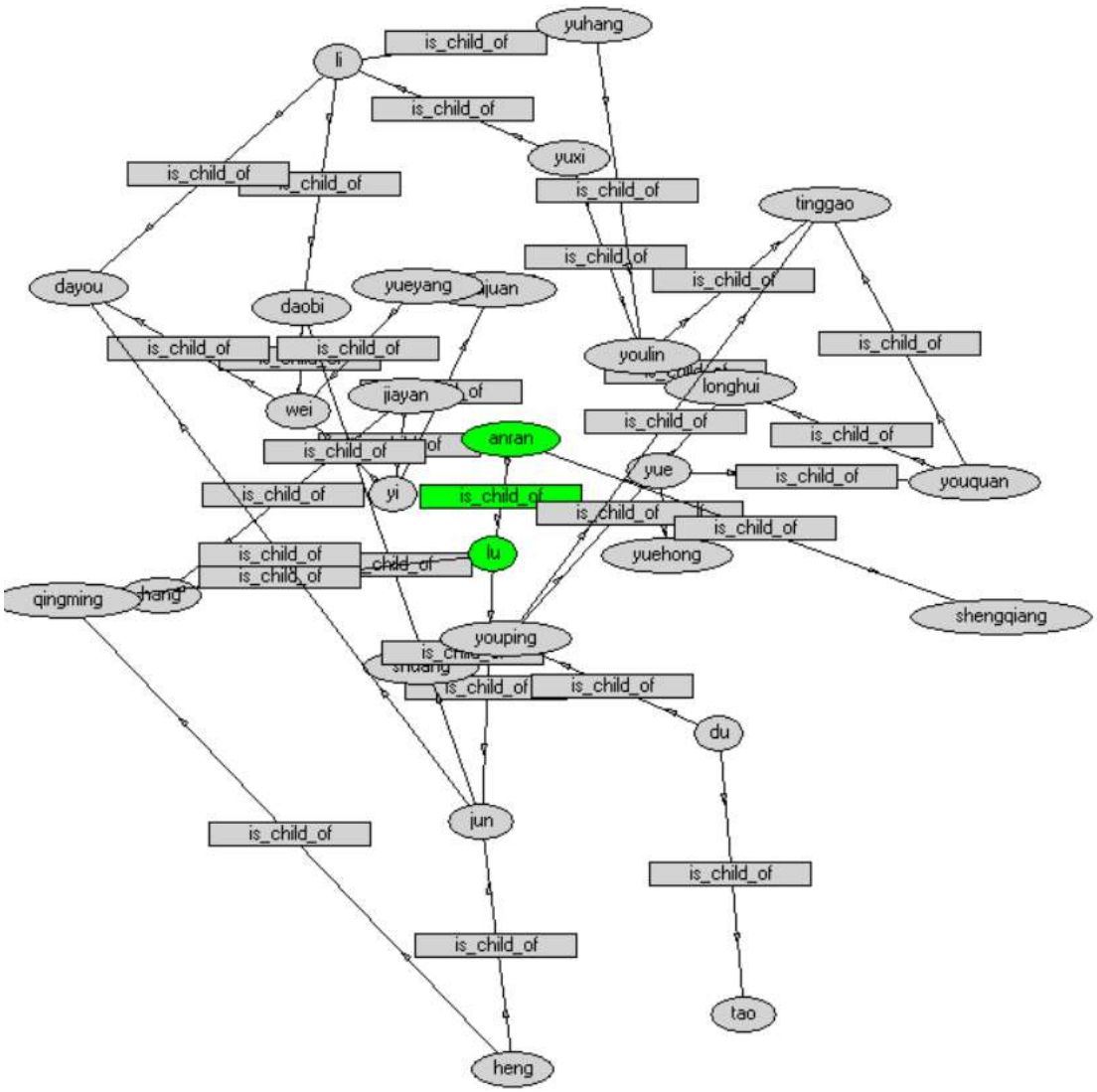
Rules definition

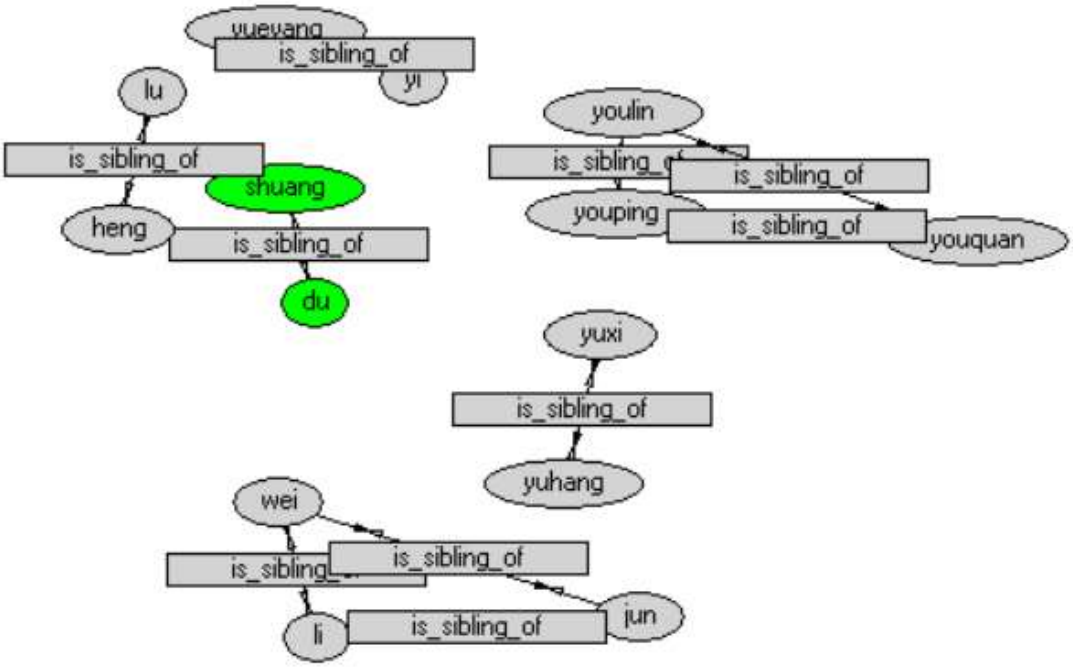
1. If x is y's father or mother, x is y's parent.
2. If y is x's father or mother, x is y's child.
3. If x is y's child, and x is male, x is y's son.
4. If x is y's child, and x is female, x is y's daughter.
5. If x is the father of y's parent, x is y's grandfather.
6. If x is the mother of y's parent, x is y's grandmother.
7. If x and y have a same parent, x and y are siblings.

8. If x is a sibling of y's parent, and x is male, x is y's uncle.
9. If x is a sibling of y's parent, and x is female, x is y's aunt.
10. If x is a child of a sibling of y's parent, x is y's cousin.
11. If x and y have a same child, x and y are spouses.
12. If a is a b, and b has attribute c, a has attribute a: c.
13. If a's hobby is the same as b's hobby, a has common hobby with b.
14. If a's hobby is different from b's hobby, a has different hobby with b.

Screenshots




yuhang
li
is_child_of
is_child_of
yuxi
is_child_of
is_child_of
is_child_of
tinggao
is_child_of
is_child_of
dayou
yueyang
daobi
is_child_of
is_child_of
youlin
is_child_of
longhui
wei
jiayan
is_child_of
is_child_of
anran
is_child_of
yue
is_child_of
youquan
yi
is_child_of
is_child_of
is_child_of
lu
yuehong
qingming
shengqiang
youping
is_child_of
is_child_of
du
jun
is_child_of
is_child_of
is_child_of
tao
heng


yueyang
is_sibling_of
yi
lu
is_sibling_of
shuang
heng
is_sibling_of
du
youlin
is_sibling_of
youping
is_sibling_of
youquan
yuxi
is_sibling_of
yuhang
wei
is_sibling_of
is_sibling_of
li
is_sibling_of
jun

Other relations are not suitable for showing, they are too complex in semantic network.

Summary

I have translated prolog into semantic environment, and add one attribute to one ontology with three rules. And try to figure out the differences between them.

The main difference between prolog and semantic is ontology. Prolog can only describe the relations between two entities. However, semantic can define the attributes of entity using ontology. What's more, semantic can also use attributes to define the relations between entities, which can't be done by prolog.