```prolog
:- use_module('metagol').

%% first-order background knowledge
mother(ann,amy).
mother(ann,andy).
mother(amy,amelia).
mother(linda,gavin).
father(steve,amy).
father(steve,andy).
father(gavin,amelia).
father(andy,spongebob).


%% predicates that can be used in the learning
prim(mother/2).
prim(father/2).

%% metarules
metarule([P,Q],([P,A,B]:-[[Q,A,B]])).
metarule([P,Q,R],([P,A,B]:-[[Q,A,B],[R,A,B]])).
metarule([P,Q,R],([P,A,B]:-[[Q,A,C],[R,C,B]])).

%% learning task
a :-
    %% positive examples
    Pos = [
        grandparent(ann,amelia),
        grandparent(steve,amelia),
        grandparent(ann,spongebob),
        grandparent(steve,spongebob),
        grandparent(linda,amelia)
    ],
    %% negative examples
    Neg = [
        grandparent(amy,amelia)
    ],
    learn(Pos,Neg).
```

```prolog
?- a.
% learning grandparent/2
% clauses: 1
% clauses: 2
% clauses: 3
grandparent(A,B):-grandparent_1(A,C),grandparent_1(C,B).
grandparent_1(A,B):-father(A,B).
grandparent_1(A,B):-mother(A,B).
true
```