Logic Programming

Query Examples

Michael Genesereth
Computer Science Department
Stanford University
Kinship

Blocks World

Map Coloring

Sierra
Kinship
parent(art, bob)
parent(art, bea)
parent(bob, cal)
parent(bob, cam)
parent(bea, cat)
parent(bea, coe)
Goal

$$\text{goal}(X,Z) :- \text{parent}(X,Y) \land \text{parent}(Y,Z)$$

Dataset:

- parent(art,bob)
- parent(art,bea)
- parent(bob,cal)
- parent(bob,cam)
- parent(bea,cat)
- parent(bea,coe)

Result:

- goal(art,cal)
- goal(art,cam)
- goal(art,cat)
- goal(art,coe)
Query:

\[
\text{goal}(X) :- \text{parent}(X,Y) \\
\text{goal}(X) :- \text{parent}(Y,X)
\]

Dataset:

<table>
<thead>
<tr>
<th>parent(art,bob)</th>
<th>goal(art)</th>
</tr>
</thead>
<tbody>
<tr>
<td>parent(art,bea)</td>
<td>goal(bob)</td>
</tr>
<tr>
<td>parent(bob,cal)</td>
<td>goal(bea)</td>
</tr>
<tr>
<td>parent(bob,cam)</td>
<td>goal(cal)</td>
</tr>
<tr>
<td>parent(bea,cat)</td>
<td>goal(cam)</td>
</tr>
<tr>
<td>parent(bea,coe)</td>
<td>goal(coe)</td>
</tr>
</tbody>
</table>
Query

\[ \text{goal}(Y,Z) : \neg \text{parent}(X,Y) \land \text{parent}(X,Z) \land \text{distinct}(Y,Z) \]

<table>
<thead>
<tr>
<th>Dataset:</th>
<th>Result:</th>
</tr>
</thead>
<tbody>
<tr>
<td>parent(art,bob)</td>
<td>goal(bob,bea)</td>
</tr>
<tr>
<td>parent(art,bea)</td>
<td>goal(bea,bob)</td>
</tr>
<tr>
<td>parent(bob,cal)</td>
<td>goal(cal,cam)</td>
</tr>
<tr>
<td>parent(bob,cam)</td>
<td>goal(cam,cal)</td>
</tr>
<tr>
<td>parent(bea,cat)</td>
<td>goal(cat,coe)</td>
</tr>
<tr>
<td>parent(bea,coe)</td>
<td>goal(coe,cat)</td>
</tr>
</tbody>
</table>
Dataset:

parent(art,bob)
parent(art,bea)
parent(art,ben)
parent(bob,eli)

Query: find every person with at least one child
Dataset:

- parent(art,bob)
- parent(art,bea)
- parent(art,ben)
- parent(bob,eli)

Query: find every person with at least one child

\[ \text{goal}(X) \leftarrow \text{parent}(X,Y) \]
Dataset:
parent(art,bob)
parent(art,bea)
parent(art,ben)
parent(bob,eli)

Query: find every person with at least two children
Dataset:

- parent(art,bob)
- parent(art,bea)
- parent(art,ben)
- parent(bob,eli)

Query: find every person with at least two children

\[
goal(X) :- \\
\text{parent}(X,Y) \land \text{parent}(X,Z) \land \text{distinct}(Y,Z)
\]
Dataset:

  parent(art,bob)
  parent(art,bea)
  parent(art,ben)
  parent(bob,eli)

Query: find every person with at least three children

\[
\text{goal}(X) :-
\begin{align*}
\text{parent}(X,Y) & \land \text{parent}(X,Z) & \land \text{parent}(X,W) \\
\text{mutex}(Y,Z,W)
\end{align*}
\]
Dataset:

parent(art,bob)
parent(art,bea)
parent(art,ben)
parent(bob,eli)

Query: find every person with exactly three children

goal(X) :-
    parent(X,Y) &
evaluate(countofall(Z,parent(X,Z)),3)

Blocks World
Blocks World
Symbols: $a, b, c, d, e$

Unary Predicate:
   block

Binary Predicate:
   on - pairs of blocks in which first is on the second
Data

block(a)
block(b) on(a,b)
block(c) on(b,c)
block(d) on(d,e)
block(e)
Blocks World - cluttered

- block(a)
- block(b) on(a,b)
- block(c) on(b,c)
- block(d) on(d,e)
- block(e)

goal(Y) :- on(X,Y)

- goal(b)
- goal(c)
- goal(e)
Blocks World - clear

```
block(a)
block(b)   on(a,b)
block(c)   on(b,c)
block(d)   on(d,e)
block(e)

goal(Y) :- block(Y) & countofall(X, on(X,Y), 0)

goal(a)
goal(d)
```
Blocks World - supported

- block(a)
- block(b)  on(a,b)
- block(c)  on(b,c)
- block(d)  on(d,e)
- block(e)

???

goal(a)
goal(b)
goal(d)
Blocks World - supported

\[
\text{block(a)} \\
\text{block(b)} \quad \text{on(a,b)} \\
\text{block(c)} \quad \text{on(b,c)} \\
\text{block(d)} \quad \text{on(d,e)} \\
\text{block(e)}
\]

\[
\text{goal(X)} :\ = \ \text{on(X,Y)}
\]

\[
\text{goal(a)} \\
\text{goal(b)} \\
\text{goal(d)}
\]
blocks World - table

block(a)
block(b) on(a,b)
block(c) on(b,c)
block(d) on(d,e)
block(e)

???

goal(c)
goal(e)
Blocks World - table

\[
\begin{align*}
\text{goal}(X) & :\text{ block}(X) \land \text{countofall}(Y, \text{on}(X,Y), 0) \\
\text{goal}(c) & \\
\text{goal}(e) &
\end{align*}
\]
Blocks World - stack

\[
\text{goal}(X,Y,Z) \leftarrow \text{on}(X,Y) \land \text{on}(Y,Z)
\]

\[
\text{goal}(a,b,c)
\]
Blocks World - above

\[
\begin{align*}
\text{block}(a) & \quad \text{on}(a,b) \\
\text{block}(b) & \quad \text{on}(b,c) \\
\text{block}(c) & \quad \text{on}(d,e) \\
\text{block}(d) & \quad \text{on}(e) \\
\end{align*}
\]

\[
\begin{align*}
goal(X,Y) & : \text{on}(X,Y) \\
goal(X,Z) & : \text{on}(X,Y) \land \text{on}(Y,Z) \\
goal(X,W) & : \text{on}(X,Y) \land \text{on}(Y,Z) \land \text{on}(Z,W) \\
\ldots
\end{align*}
\]

\[
\begin{align*}
goal(a,b) \\
goal(b,c) \\
goal(a,c) \\
goal(d,e)
\end{align*}
\]
Map Coloring
Dataset

hue(red)
hue(green)
hue(blue)
hue(purple)
\texttt{goal(C1,C2,C3,C4,C5,C6) :-}
\texttt{hue(C1) & hue(C2) & hue(C3) & hue(C4) & hue(C5) & hue(C6) &}
\texttt{distinct(C1,C2) & distinct(C1,C3) & distinct(C1,C5) &}
\texttt{distinct(C1,C6) & distinct(C2,C3) & distinct(C2,C5) &}
\texttt{distinct(C2,C6) & distinct(C3,C4) &}
\texttt{distinct(C3,C6) & distinct(C5,C6)}
Example
Example

SEND
+MORE
-----
MONEY
goal(S,E,N,D,M,O,R,Y) :-
digit(S) & digit(E) & digit(N) & digit(D) &
digit(M) & digit(O) & digit(R) & digit(Y) &
M!=0 & M!=S & M!=E & M!=N & M!=D &
O!=S & O!=E & O!=N & O!=D & O!=M &
evaluate(S*1000+E*100+N*10+D,X) &
evaluate(M*1000+O*100+R*10+E,Y) &
evaluate(M*10000+O*1000+N*100+E*10+Y,Z) &
evaluate(plus(X,Y),Z)
Computational Analysis

Data

digit(1)  digit(6)
digit(2)  digit(7)
digit(3)  digit(8)
digit(4)  digit(9)
digit(5)  digit(0)

Rule

goal(S,E,N,D,M,O,R,Y) :-
  digit(S) & digit(E) & digit(N) & digit(D) &
  digit(M) & digit(O) & digit(R) & digit(Y) & ...

Analysis

10x10x10x10x10x10x10x10 = 10^8 = 100,000,000 cases

111,111,110 unifications

Running time ~ minutes