Logic-Based Learning: Learning from Answer Sets and ILASP

Mark Law

February 16, 2015

Question 1

For each of set of Mode declarations below, calculate a maximal search space

1. $M = \begin{cases} modeh(1, p) \\ modeh(1, q(var(t))) \\ modeb(2, r(var(t))) \end{cases}$

   ($B_{max} = 2$, $H_{min} = H_{max} = 2$, $V_{max} = 2$).

2. $M = \begin{cases} modeh(1, p) \\ modeh(1, q(var(t))) \\ modeb(2, r(var(t))) \\ modeb(2, \text{not } s(var(t))) \end{cases}$

   ($B_{max} = 2$, $H_{min} = H_{max} = 2$, $V_{max} = 2$).

3. (BIG)

   $M = \begin{cases} modeh(1, serve(const(food), var(person))) \\ modeb(1, serve(var(food)), \\ modeb(1, var(person))), \\ modeb(1, vegetarian(var(person))), \\ modeb(1, meat(var(food))), \\ modeb(1, person(var(person))) \end{cases}$

   ($H_{min} = 4$, $H_{max} = 4$, $V_{max} = 2$). There is an additional restriction that only 1 variable can appear in the head of any rule and the upper/lower bounds of any aggregates must be less than or equal to 1.

Question 2

In this question, we only consider normal rules in our search space. For each of the following ILP_{LAS} tasks.

i) $$B = \begin{cases} bird(a). \\ bird(b). \\ 1\{can(a, fly), can(a, swim)\}1. \\ can(b, fly). \\ ability(swim).ability(fly). \end{cases}$$

$$E^+ = \{ \ \langle\{penguin(a)\}, \emptyset\rangle$$

$$E^- = \{ \ \langle\{penguin(b)\}, \emptyset\rangle$$

$$M = \begin{cases} modeh(1, penguin(var(bird))) \\ modeb(1, bird(var(bird))) \\ modeb(1, \text{not } can(var(bird), const(ability))) \end{cases}$$

ii) $$B = \begin{cases} r \leftarrow q. \\ r \leftarrow p. \\ 0\{a\}1. \end{cases}$$

$$E^+ = \begin{cases} \langle\{p\}, \{q\rangle, \\ \langle\{q\}, \{p\}\rangle, \\ \langle\{a\}, \emptyset\rangle, \\ \langle\emptyset, \{a\}\rangle. \end{cases}$$

$$E^- = \begin{cases} \langle\{a\}, \{r\}\rangle, \\ \langle\{r\}, \{a\}\rangle. \end{cases}$$

$$M = \begin{cases} modeh(1, p) \\ modeh(1, q) \\ modeb(1, \text{not } p) \\ modeb(1, \text{not } q) \\ modeb(1, a) \end{cases}$$

iii) (HARD)
$$B = \{ \ p(c, d).$$

$$E^+ = \begin{cases} \langle\emptyset, \{q(c, d)\}\rangle \\ \langle\{q(c, d)\}, \emptyset\rangle \end{cases}$$

$$E^- = \emptyset$$

$$M = \begin{cases} modeh(1, q(var(any), var(any))) \\ modeb(1, p(var(any), var(any))) \\ modeb(1, \text{not } q(var(any, var(any))) \end{cases}$$

Only consider rules which use each predicate once (or not at all) in the body.

a) Write down the search space with $V_{max} = 2$ and $B_{max} = 2$ (only consider normal rules in this search space).

b) Find an optimal inductive solution

c) Calculate the positive solutions for size 1 to 6

d) Calculate the violating solutions for size 1 to 6.

e) Calculate the inductive solutions for size 1 to 6.

Question 3

Translate the following ILP tasks into ILP_{LAS} tasks (give the positive and negative examples) and then use ILASP with the mode declarations below to find an optimal inductive solution for them.

$$M = \left\{ \begin{array}{l} modeh(1, s) \\ modeh(1, q) \end{array} \right.$$

i) ILP_b: $B = \langle \{p \leftarrow \text{not } q; q \leftarrow \text{not p}; r \leftarrow \text{not } s; s \leftarrow p\}$, $E^+ = \{q\}$, $E^- = \{p, r\}$.

ii) ILP_c: $B = \langle \{p \leftarrow \text{not } q; q \leftarrow \text{not p}; r \leftarrow \text{not } s; s \leftarrow p\}$, $E^+ = \{q\}$, $E^- = \{p, r\}$.

Question 4

Use ILASP to find the set of optimal inductive solutions for each of the following tasks: (The first one has been written using the ILASP syntax and is already available on the website).

1.
```
#background.
p :- not q.
q :- not p.
#pos.
q
not r
#pos.
q
r
#pos.
p
#neg.
p
r
#mode_declarations.
modeh(1, p).
modeh(1, q).
modeh(1, r).
```

```
modeh(1, s).
modeb(1, p).
modeb(1, q).
modeb(1, r).
modeb(1, s).
```

2. $B = \left\{ \begin{array}{l} r \leftarrow q. \\ r \leftarrow p. \\ 0\{a\}1. \end{array} \right.$

   $E^+ = \left\{ \begin{array}{l} \langle\{p\}, \{q\rangle, \\ \langle\{q\}, \{p\}\rangle, \\ \langle\{a\}, \emptyset\rangle, \\ \langle\emptyset, \{a\}\rangle. \end{array} \right.$

   $E^- = \left\{ \begin{array}{l} \langle\{a\}, \{r\}\rangle, \\ \langle\{r\}, \{a\}\rangle. \end{array} \right.$

   $M = \left\{ \begin{array}{l} modeh(1, p) \\ modeh(1, q) \\ modeb(1, \text{not } p) \\ modeb(1, \text{not } q) \\ modeb(1, a) \end{array} \right.$

3. (HARD)
   $B = \{\ p(c, d).$

   $E^+ = \left\{ \begin{array}{l} \langle\emptyset, \{q(c, d)\}\rangle \\ \langle\{q(c, d)\}, \emptyset\rangle \end{array} \right.$

   $E^- = \emptyset$

   $M = \left\{ \begin{array}{l} modeh(1, q(var(any), var(any))) \\ modeb(1, p(var(any), var(any))) \\ modeb(1, \text{not } q(var(any, var(any))) \end{array} \right.$

Note that the implementation of ILASP on the course website treats $modeb(r, s)$ as $modeb(r, s)$ and $modeb(r, \text{not } s)$. You do not need to specify that the literal can also be negative.

Question 5

Consider the Learning from Entailment task:

$B = \left\{ \begin{array}{l} parent(mike, mark) \\ parent(sue, mark) \\ parent(howell, sue) \\ parent(norma, sue) \end{array} \right.$

$$E^+ = \begin{cases} ancestor(mike, mark), \\ ancestor(sue, mark), \\ ancestor(howell, mark), \\ ancestor(norma, mark), \\ ancestor(howell, sue) \end{cases}$$

$$E^- = \{ancestor(mark, mark)$$

$$M = \begin{cases} modeh(1, ancestor(var(any), var(any))) \\ modeb(1, ancestor(var(any), var(any))) \\ modeb(1, parent(var(any), var(any))) \end{cases}$$

Note that for any definite logic program P, P's single answer set is equal to the minimal Herbrand model of the program $M(P)$. Use this property to represent the problem as an ILASP task.

Note that this problem shows that ILASP is capable of learning recursive definitions; in fact, so long as $B \cup S_M$ grounds finitely and ignoring constructs such as lists which ASP does not have, ILASP is sound and complete with respect to Learning from Entailment tasks and can solve any of the tasks shown previously in the course.

Advanced Questions

Question 6

Prove that for any ILP_{LAS} task, the inductive solutions of any given length are exactly those positive solutions which are not violating solutions.