

Inteligência Artificial – IA

2º Trabalho Prático

Lucas Castro, Suziane Cundiff, Marcos Guerreiro

1 Experimento

Seguindo as especificações do trabalho prático, implementamos o indutor Hyper presente no livro Prolog Programming for Artificial Intelligence. Outra parte necessária para o experimento foi a implementação do programa Learning about arch. Porém, nessa parte a implementação do indutor não funcionou de maneira esperada. Retornando o seguinte resultado abaixo:

```
Refining hypo with cost 102:
arch(A,B,C) :- .
Hypotheses generated: 17
Hypotheses refined: 4
To be refined: 12
true .
```

Realizamos uma comparação com a implementação da equipe com a implementação disponibilizado no livro e não encontramos diferenças. Além dos predicados prove/3 que não estavam presentes no livro, e por esse motivo utilizamos o mesmo prove/3 do mini.hyper que funciona corretamente.

O indutor se encontra em *hyper.pl* e a knowledge base com as informações de indução dos arcos se encontram em *knowledge_bases/arch.pl*.

Para executar o indutor digitar os seguintes comandos:

```
$ prolog -s hyper.pl knowledge_bases/arch.pl.
$ induce(H)
```

2 Aprendizado por indução em programação lógica

É uma forma de aprendizagem de máquina onde são dados alguns exemplos da tarefa que deve ser executada (exemplo *knowledge_bases/arch.pl*. incorretos) e um programa é criado automaticamente para resolver a tarefa. Por exemplo, podem ser dados alguns exemplos de listas de tamanho par, exemplos de listas

de tamanho ímpar e automaticamente é gerado um programa que é capaz de realizar a tarefa de que é o conhecimento que se tem antes de iniciar o aprendizado, é especificado de maneira bem geral e por isso o preço por essa flexibilidade é, geralmente, uma complexidade computacional elevada para resolver as tarefas.

O programa HYPER (Hypothesis Refiner ou Refinador de hipóteses) usado nesse trabalho, é capaz de resolver exercícios de aprendizagem típicos no ambiente ILP.

Programação em lógica indutiva é similar a mineração de dados já que estamos interessados em encontrar padrões no nosso banco de conhecimento. Além de que, nos oferece um tipo de estrutura que pode ser utilizada em qualquer banco de conhecimento. Vale notar também que a programação lógica indutiva não depende de um grande número de exemplos. Pelo fato de o conhecimento aprendido por programação em lógica é por base de regras e é mais fácil de ser entendida, examinada e estendida por seres humanos.

A programação em lógica indutiva é bastante utilizada para aplicações de bioinformática e processamento de linguagem natural. Foi introduzida teoricamente pela primeira vez por Gordon Plotkin e Ehudshapro. Em 1981, eles implementaram o primeiro exemplo de programa lógico a partir de exemplos positivos e negativos. Após isso, em 1991, foi fundada a Conferência Internacional sobre Programação em Lógica por Muggleton.

Da forma que foi apresentada no livro Prolog Programming for Artificial Intelligence, a programação em lógica indutiva se baseia em partir de regras mais gerais que abranjam todos os exemplos positivos, para regras mais específicas, que ainda abranjam todos os exemplos positivos mas também abrange menos exemplos negativos que a regra anterior. O livro discute duas formas de fazer as regras serem mais específicas, unificando variáveis diferentes nas regras ou adicionando novos predicados à regra atual. Dessa forma, se garante que as regras fiquem mais específicas já que a nova regra é válida somente para um subconjunto das regras anteriores.

No livro também são discutidas diversas formas de diminuir o espaço de busca do indutor já que o espaço de busca para esse tipo de programa costuma ser bem grande. Algumas formas de diminuir o espaço de busca são:

- utilizar tipagem nas variáveis das cláusulas, para evitar que variáveis de tipos diferentes sejam unificadas;
- lidar com termos estruturados;
- utilizar busca em largura em vez de busca em profundidade iterativa para encontrar a resposta com menor número de predicados de forma mais rápida e eficiente;

- distinção entre variáveis de entrada e de saída nos predicados já que as variáveis de saídas nunca podem estar estanciadas na chamada do predicado.

Dessa forma concluímos que a programação em lógica indutiva é uma forma de aprendizagem indutiva que utiliza programação em lógica e pode ser utilizada para resolver problemas de aprendizagem de máquina. E tem como vantagem ter como resultado regras que podem ser mais facilmente interpretadas por outros seres humanos que outras abordagens, além de precisar de poucos exemplos para ser efetiva.