

Universidade Federal do Amazonas
Mestrado e Doutorado Interinstitucional

Prolog

Aulas 07-08

Inteligência Artificial
2016/1

José Francisco de Magalhães Netto

jnetto@icomp.ufam.edu.br

Boa Vista, 12/04/2016

Créditos

Baseado no Material Disponibilizado pela
Prof^a Andrea Schwertner Charão
[http://www-usr.inf.ufsm.br/~andrea/elc117/prolog-
listas.pdf](http://www-usr.inf.ufsm.br/~andrea/elc117/prolog-listas.pdf)

Lógica e Programação de Computadores

- Na lógica de predicados usamos regras de inferência para demonstrar que uma tese é consequência de determinadas hipóteses
- Programação em Lógica e especificamente a linguagem **Prolog** – *Progamming in Logic* – também pode provar teses a partir de hipóteses
- A linguagem Prolog inclui: predicados, conectivos lógicos e regras de inferência - Princípio da Resolução

Silogismo em Prolog

Silogismo

Sócrates é um homem.

Todos os homens são mortais.

Sócrates é mortal?

Prolog

homem(socrates).

mortal(X) :- homem(X).

?- mortal(socrates).

Lógica e Programação de Computadores

- Prolog é uma linguagem declarativa ao invés de procedimental. Isto significa que ela especifica **o quê** se sabe e **o que** deve ser feito, mas não especifica **como** fazer.
- Um programa Prolog consiste na declaração (ou descrição de uma interpretação) de *hipóteses que são verdadeiras em uma interpretação*

Lógica e Programação de Computadores

- O conjunto de declarações que formam um programa Prolog é chamado de base de conhecimento (BC) ou base de fatos.
- Para determinar se uma tese (consulta do usuário à BC) é ou não verdadeira, o Prolog aplica suas regras de inferência na BC sem a necessidade de instruções adicionais por parte do programador.

Lógica e Programação de Computadores

- Bases de conhecimento convencionais descrevem apenas fatos
“Tico é um pinguim”
- As sentenças de um Programa em Lógica, além de descrever fatos, permitem a descrição de regras
“Todo pinguim é um ave”
- Havendo regras, novos fatos podem ser deduzidos
“Tico é uma ave”

Lógica e Programação de Computadores

- As sentenças de um programa Prolog são expressas por cláusulas
- Tipos de cláusulas: fatos e regras
 - **Fato**: declaração de uma verdade incondicional
 - **Regra**: condição que deve ser satisfeita para que uma declaração seja considerada verdadeira

A Linguagem Prolog

- Programar em Prolog consiste em:
 - Declarar alguns **fatos** sobre objetos e suas relações
 - Definir algumas **regras** sobre objetos e suas relações
 - Fazer **consultas** sobre objetos e suas relações

A Linguagem Prolog

- Exemplo Prático: Representando Relações Familiares

Programando em SWI_Prolog

Resolvido no quadro-branco

Minimundo

Paulo é casado com Joana. Eles tem três filhos: Bruna, Carlos e Patrícia. Pedro é irmão de Paulo. Joana é irmã de Tatiana. Patrícia tem uma filha chamada Juliana.

A Linguagem Prolog

casado(paulo, joana).

pai(paulo, carlos).

pai(paulo, bruna).

pai(paulo, patricia).

irmao(pedro, paulo).

mae(joana, bruna).

A Linguagem Prolog

```
casado(paulo, joana).  
pai(paulo, carlos).  
pai(paulo, bruna).  
pai(paulo, patricia).  
irmao(pedro, paulo).  
mae(joana, bruna).  
tio(X,Z) :- irmao(X,Y),  
            pai(Y,Z).  
tio(X,Z) :- irmao(X,Y),  
            mae(Y,Z).
```

A Linguagem Prolog

FATOS

- Os objetos (argumentos) devem ser colocados dentro de parênteses e serem separados por vírgula. Ex:
 - homem(pedro).
 - brigou(carlos, ana).
 - pai(pedro, ana).
- **Aridade** é o termo usado para determinar a quantidade de objetos que o argumento de um predicado possui.
- Todo predicado é terminado com o ponto final.
- A ordem dos objetos é importante:
 - gosta(maria, josé). \neq gosta(josé, maria).

A Linguagem Prolog

FATOS

Exemplo: um sistema ecológico para especificar a cadeia alimentar

come (urso, peixe).

come (urso, raposa).

come (cavalo, mato).

animal (urso).

animal (peixe).

animal (raposa).

animal (cavalo).

% predicado binário

% predicado unário

A Linguagem Prolog

CONSULTAS

- De posse do programa Prolog (BC), pode-se fazer consultas.
- Uma consulta começa com um ponto de interrogação seguido do sinal de menos e termina com ponto final. Exemplo usando os fatos do slide anterior:
?- pai(pedro, ana).
True.

?- pai(pedro, carlos).
False.
- Quando uma consulta é feita, o Prolog realiza uma busca na sua base de conhecimento, procurando um fato que se unifique à questão.

A Linguagem Prolog

CONSULTAS

- Dois fatos se unificam se:
 - Os nomes dos seus predicados são os mesmos.
 - Eles possuem o mesmo número de argumentos.
 - Os argumentos são iguais.
- Se o Prolog encontra um fato que se unifique à pergunta, ele retorna True, caso contrário, ele retorna False.

A Linguagem Prolog

CONSULTAS e VARIÁVEIS

- Um consulta em Prolog pode usar variáveis no intuito de descobrir valores como resposta ao invés de resposta apenas do tipo True e False
- Uma variável em Prolog SEMPRE começa com letra maiúscula.
- Uma variável pode estar instanciada ou não-instanciada. Ela está instanciada quando assume o valor de um objeto e, caso contrário, está não-instanciada.
 - ?- pai (pedro,X).
 - X = ana.
 - True

A Linguagem Prolog

CONSULTAS e VARIÁVEIS

- Usando a BC do slide anterior, pode-se fazer os seguintes questionamentos:

?- come (cavalo, mato).

True

?- come (urso, coelho).

False

?- come (urso, X).

X = peixe ;

raposa

A Linguagem Prolog

REGRAS

- São utilizadas para construir relações entre fatos.
- As regras especificam coisas que podem ser verdadeiras se algumas condições forem satisfeitas.
- As regras possuem duas partes:
 - CORPO, que define as condições e se encontra na parte direita da regra.
 - CABEÇA, que define a conclusão e se encontra na parte esquerda da regra.
- A cabeça e o corpo são separados pelo símbolo :- (que é lido como se).
- Uma regra SEMPRE é terminada com o ponto final.

A Linguagem Prolog

REGRAS

- Exemplo de uma regra em Prolog:
gosta(joão, X) :- gosta(X, vinho).
- Utiliza-se a vírgula (,) no lugar da conjunção (E) e o ponto e vírgula (;) no lugar da disjunção (OU).

```
irmão(X,Y) :- pai(H,X), pai(H,Y);  
             mãe(M,X), mãe(M,Y).
```

Ou pode-se criar duas regras:

```
irmão(X,Y) :- pai(H,X), pai(H,Y).  
irmão(X,Y) :- mãe(M,X), mãe(M,Y).
```

A Linguagem Prolog

ESTRUTURAS

- Estruturas são objetos de dados que possuem uma quantidade fixa de componentes, cada um deles podendo ser acessado individualmente.

Ex.: `data(6,agosto,2003)`.

Os componentes são 6, agosto e 2003

- Para combinar os componentes de uma estrutura usamos um functor. Nesse exemplo, o functor é a palavra `data`.

A Linguagem Prolog

REGRAS e CONSULTAS

- Acrescentando uma nova regra à BD do Slide 10, pode-se fazer um novo tipo de consulta

come (urso, peixe).

come (urso, raposa).

% predicado binário

come (cavalo, mato).

animal (urso).

animal (peixe).

% predicado unário

animal (raposa).

animal (cavalo).

presa(X) :- come(Y,X), animal(X). *% regra nova*

?- presa(QUEM).

QUEM = peixe ; raposa

A Linguagem Prolog

REGRAS e LÓGICA

- Uma **regra** é a descrição de um predicado através de uma implicação em lógica
- Exemplo: “um animal é presa se é comido por outro animal”.

$\text{come}(Y,X) \wedge \text{animal}(X) \rightarrow \text{presa}(X)$

em Prolog:

$\text{presa}(X) \text{ :- come}(Y,X), \text{animal}(X)$

A Linguagem Prolog

EXEMPLO DE PROGRAMA E CONSULTAS

```
come (urso, peixe).  
come (peixe,peixinho).  
come (peixinho,alga).  
come (quati,peixe).  
come(urso,quati).  
come (urso, raposa).  
come(raposa,coelho).  
come (coelho, mato).  
come(urso,cavalo).  
come(cavalo,mato).  
come(gato-selvagem,cavalo).
```

```
animal(urso).  
animal(peixe).  
animal(peixinho).
```

```
animal(quati).  
animal(raposa).  
animal(coelho).  
animal(cavalo).  
animal(gato-selvagem).
```

```
planta(mato).  
planta(alga).
```

```
presa(X) :- come(Y,X), animal(X).  
caçado(X) :- presa(X).
```


A Linguagem Prolog

EXEMPLO DE PROGRAMA E CONSULTAS

- Consultas e respostas:

?- animal(coelho).

true

?- come(gato_selvagem,mato).

false

?- come(X,peixe).

urso;

quati.

?- come(X,Y),planta(Y).

peixinho alga;

coelho mato;

cavalo mato.