



**Data Science  
Academy**

[www.datascienceacademy.com.br](http://www.datascienceacademy.com.br)

Introdução à Inteligência Artificial

Encadeamento

Um algoritmo de encadeamento possui um conceito simples: começar com as sentenças atômicas da base de conhecimento e aplicar uma regra de inferência (como Modus Ponens) no sentido para a frente, acrescentando novas sentenças atômicas até não ser mais possível fazer nenhuma inferência adicional. Aqui, explicamos como o algoritmo é aplicado a cláusulas definidas de primeira ordem. Cláusulas definidas como Situação Resposta são especialmente úteis no caso de sistemas que fazem inferências em resposta a informações recém-chegadas. Muitos sistemas podem ser definidos desse modo, e o encadeamento para a frente pode ser implementado muito eficientemente.

As cláusulas definidas de primeira ordem são muito semelhantes às cláusulas definidas proposicionais: elas são disjunções de literais dos quais exatamente um é positivo. Uma cláusula definida é atômica ou é uma implicação cujo antecedente é uma conjunção de literais positivos e cujo consequente é um único literal positivo. As cláusulas a seguir são cláusulas definidas de primeira ordem:

$$Rei(x) \wedge Ambicioso(x) \Rightarrow Perverso(x).$$

$$Rei(João).$$

$$Ambicioso(y).$$

Ao contrário dos literais proposicionais, os literais de primeira ordem podem incluir variáveis e, nesse caso, essas variáveis são consideradas universalmente quantificadas (em geral, omitimos os quantificadores universais quando escrevemos cláusulas definidas).

Nem toda base de conhecimento pode ser convertida em um conjunto de cláusulas definidas, devido à restrição de único literal positivo, mas muitas podem. Considere o problema a seguir:

A lei diz que é crime um americano vender armas a nações hostis. O país “Nono” (nome fictício), inimigo da América, tem alguns mísseis, e todos foram vendidos pelo Coronel West, um americano. Provaremos que West é um criminoso. Primeiro, vamos representar esses fatos como cláusulas definidas de primeira ordem. Na sequência mostraremos como o algoritmo de encadeamento para a frente resolve o problema.

“... é crime um americano vender armas a nações hostis”:

$$Americano(x) \wedge Arma(y) \wedge Vende(x, y, z) \wedge Hostil(z) \Rightarrow Criminoso(x).$$

“Nono... tem alguns mísseis.” A sentença:  $x \text{ Possui}(\text{Nono}, x) \wedge \text{Míssil}(x)$  é transformada em duas cláusulas definidas por eliminação existencial, introduzindo-se uma nova constante M1:

$$\text{Possui}(\text{Nono}, M_1)$$
$$\text{Míssil}(M_1)$$

“Todos foram vendidos pelo Coronel West”:

$$\text{Míssil}(x) \wedge \text{Possui}(\text{Nono}, x) \Rightarrow \text{Vende}(\text{West}, x, \text{Nono})$$

Também precisamos saber que mísseis são armas:

$$\text{Míssil}(x) \Rightarrow \text{Arma}(x)$$

e devemos saber que um inimigo da América é considerado “hostil”:

$$\text{Inimigo}(x, \text{América}) \Rightarrow \text{Hostil}(x)$$

“West, um americano...”:

$$\text{Americano}(\text{West})$$

“O país Nono, inimigo da América...”:

$$\text{Inimigo}(\text{Nono}, \text{América})$$

Essa base de conhecimento não contém nenhum símbolo de função e, portanto, é uma instância da classe Datalog de bases de conhecimento. Datalog é uma linguagem restrita a cláusulas definidas de primeira ordem sem símbolos de funções. O nome Datalog deve-se a poder representar o tipo de asserções feitas tipicamente em bancos de dados relacionais. Veremos que a ausência de símbolos de funções torna a inferência muito mais fácil.

Aqui está um exemplo de algoritmo de encadeamento para a frente muito simples, como. Começando pelos fatos conhecidos, ele ativa todas as regras cujas premissas são satisfeitas, adicionando suas conclusões aos fatos conhecidos. O processo se repete até a consulta ser respondida (supondo-se que apenas uma resposta seja necessária) ou que nenhum fato novo seja adicionado. Note que um fato não é “novo” se for apenas uma renomeação de um fato conhecido. Uma sentença é uma renomeação de outra se elas são sentenças idênticas,

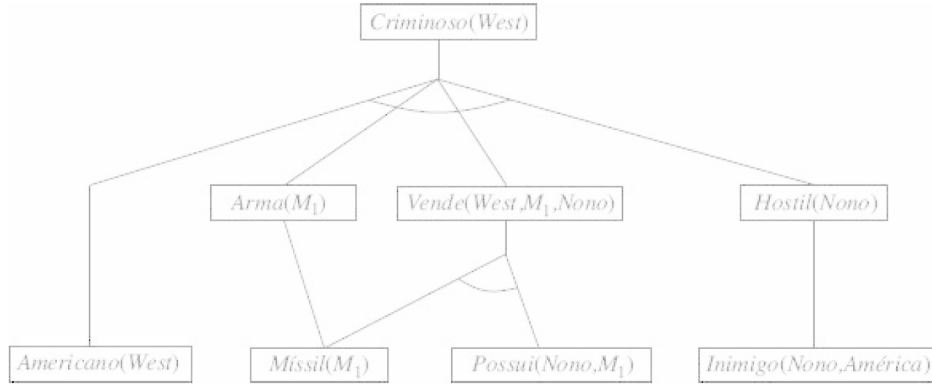
exceto pelos nomes das variáveis. Por exemplo,  $Gosta(x, Sorvete)$  e  $Gosta(y, sorvete)$  são renomeações uma da outra porque diferem apenas na escolha de  $x$  ou  $y$ ; seus significados são idênticos: todo mundo gosta de sorvete.

```
função ASK-LPO-EF( $BC, a$ ) retorna uma substituição ou falso
  entradas:  $BC$ , a base de conhecimento, um conjunto de cláusulas definidas de primeira ordem
     $a$ , a consulta, uma sentença atômica
  variáveis locais:  $nova$ , as novas sentenças deduzidas em cada iteração

  repita até  $nova$  seja vazio
     $nova \leftarrow \{ \}$ 
    para cada  $regra$  em  $BC$  faça
       $(p1 \wedge \dots \wedge pn \sqsubset q) \leftarrow \text{PADRONIZAR-VARIÁVEIS}(regra)$ 
      para cada  $q$  tal que  $\text{SUBST}(q, p1 \wedge \dots \wedge pn) = \text{SUBST}(q, p'1 \wedge \dots \wedge p'n)$ 
        para algum  $p'1, \dots, p'n$  em  $BC$ 
           $q' \leftarrow \text{SUBST}(q, q)$ 
          se  $q'$  não unifica com alguma sentença já em  $BC$  ou  $nova$  então faça
            adicionar  $q'$  a  $nova$ 
           $\varphi \leftarrow \text{UNIFICAR}(q', a)$ 
          se  $\varphi$  não é falha então retornar  $\varphi$ 
      adicionar  $nova$  a  $BC$ 
  retornar falso
```

Algoritmo conceitualmente simples, em cada iteração, ele acrescenta a  $BC$  todas as sentenças atômicas que podem ser deduzidas em uma única etapa das sentenças de implicação e das sentenças atômicas que já estão em  $BC$ . A função PADRONIZAR-VARIÁVEIS substitui todas as variáveis em seus argumentos com outras que nunca foram utilizadas antes.

A figura abaixo mostra a árvore de prova gerada. Note que nenhuma nova inferência é possível nesse ponto porque toda sentença que poderia ser uma conclusão produzida por encadeamento para a frente já está contida explicitamente na  $BC$ . Tal base de conhecimento é chamada ponto fixo do processo de inferência. Os pontos fixos alcançados por encadeamento para a frente com cláusulas definidas de primeira ordem são semelhantes aos do encadeamento para a frente proposicional; a principal diferença é que um ponto fixo de primeira ordem pode incluir sentenças atômicas universalmente quantificadas.



### Referências:

Livro: Inteligência Artificial

Autor: Peter Norvig