REPRESENTAÇÃO DE CONHECIMENTO

Luís Morgado

ISEL-ADEETC

LÓGICA DE PREDICADOS

QUANTIFICAÇÃO

A possibilidade de **generalização** obtém-se em lógica de predicados através da utilização de **quantificadores** e **variáveis** a eles associadas

Quantificador universal: "*Todos os homens são mortais*"

$$(\forall x)(homem(x) \rightarrow mortal(x))$$

Quantificador existencial: "*Qualquer pessoa tem uma mãe*"

$$(\forall x)(pessoa(x) \rightarrow (\exists y)mãe(x, y))$$

LÓGICA DE PREDICADOS

LÓGICA DE PRIMEIRA ORDEM

- **Predicado**
  - Expressão que pode ser verdade ou falsa consoante o valor das variáveis
  - $P(x)$
  - $P(x_i)$ é verdade se $x_i \in \{x \mid P(x)\}$,
- **Variável**
  - Especifica um local de uma expressão onde pode ocorrer uma **substituição** (transformação sintáctica ou formal de uma expressão)
  - **Quantificação**
    - Especificação da quantidade elementos de um domínio que satisfazem uma fórmula

LÓGICA DE PRIMEIRA ORDEM

QUANTIFICADORES

Quantificador universal

- $\forall$
- *Para qualquer elemento do domínio*
- *Para todo o elemento do domínio*

Quantificador existencial

- $\exists$
- *Para pelo menos um elemento do domínio*

Exemplo

$\forall x \in X, P(x) \vee Q(x)$

$\exists\ x \in X, P(x) \vee Q(x)$

LÓGICA DE PRIMEIRA ORDEM

QUANTIFICAÇÃO

- Uma fórmula contendo uma variável **quantificada universalmente** é avaliada em determinado domínio com valor de verdade *verdadeiro,* **se e só se for avaliada com o valor *verdadeiro* para todas as substituições da variável** por um elemento do domínio considerado.

- Uma fórmula contendo uma variável **quantificada existencialmente** é avaliada em determinado domínio com valor de verdade *verdadeiro,* **se e só se for avaliada com o valor *verdadeiro* para, pelo menos, uma substituição da variável** por um elemento do domínio considerado.

LÓGICA DE PRIMEIRA ORDEM

SUBSTITUIÇÃO,

As ocorrências das variáveis dizem-se ***livres*** ou ***ligadas***, dependendo se estão, ou não estão, no âmbito de um quantificador

A *substituição* de uma variável (livre) X por um termo θ é denotada por $[\theta / X]$ (θ substitui X)

UNIFICAÇÃO

Determinação da substituição mais geral que iguale dois termos, se existir

PRINCÍPIO DA RESOLUÇÃO

$$\left.\begin{array}{c} p(X) \vee q(X) \\ \neg p(b) \vee r(a) \\ \hline q(b) \vee r(a) \end{array}\right| \sigma = \{b / X\}$$

LÓGICA DE PRIMEIRA ORDEM

SINTAXE

- **Termos**
  - **Variáveis**
  - **Constantes**
  - **Funções**
    - Denotam transformações
- **Fórmulas**
  - Denotam asserções sobre os termos
  - **Predicados atómicos**
  - **Conectivas lógicas**
    - Se α e β forem fórmulas, então $\neg\alpha$, $\alpha \vee \beta$, $\alpha \wedge \beta$, $\alpha \rightarrow \beta$ são fórmulas
  - **Quantificadores**
    - Se α for uma fórmula e X uma variável, então $(\forall X)\alpha$ e $(\exists X)\alpha$ são fórmulas

SISTEMA FORMAL (SIMBÓLICO)

- **LINGUAGEM**
  - **Alfabeto**
    - Conjunto de símbolos
    - Expressões
      - Sequências finitas de símbolos
  - **Fórmulas** (bem formadas)
    - Expressões válidas da linguagem
- **AXIOMAS**
  - Subconjunto de fórmulas do sistema formal
- **REGRAS DE INFERÊNCIA**
  - Premissas (condições)
  - Conclusões

LÓGICA DE PRIMEIRA ORDEM

REPRESENTAÇÃO DE CONHECIMENTO

AXIOMAS

Um axioma é uma **expressão da linguagem** que constitui uma **parte de conhecimento aceite como verdade**

- Um axioma pode ser **relativo ao domínio** em questão (***axioma do domínio***)
- Um **conjunto de axiomas de um domínio** constituem uma ***teoria do domínio***

LÓGICA DE PRIMEIRA ORDEM

EXEMPLO
Representação de redes semânticas

Factos:

subclasse(*ave*, *vertebrado*)

subclasse(*mamífero*, *vertebrado*)

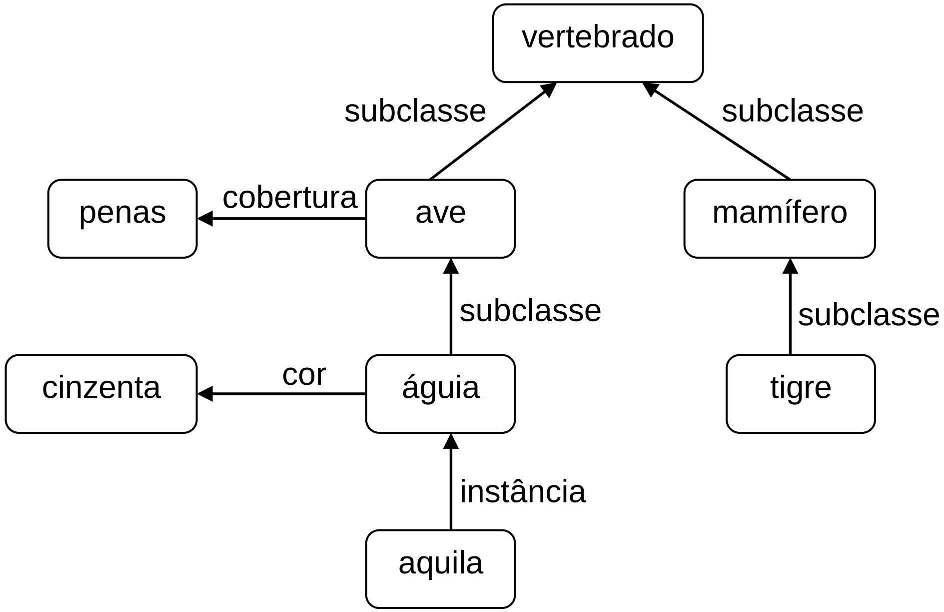
subclasse (*águia*, *ave*)

subclasse(*tigre*, *mamífero*)

instância(*aquila*, *águia*)

característica(*ave*, *cobertura*, *penas*)

característica(*águia*, *cor*, *cinzenta*)



LÓGICA DE PRIMEIRA ORDEM

EXEMPLO

Representação de redes semânticas

O conhecimento geral sobre classes e instâncias pode ser descrito da seguinte forma:

Predicados:

$instância(Objecto, Classe)$

$subclasse(Classe, SuperClasse)$

$característica(Classe, Atributo, Valor)$

Regras:

$$(\forall c)(\forall s)(\forall i)(subclasse(c,s) \wedge instancia(i,c) \rightarrow instancia(i,s))$$

$$(\forall c)(\forall s)(\forall z)(subclasse(c,s) \wedge subclasse(s,z) \rightarrow subclasse(c,z))$$

$$(\forall c)(\forall s)(\forall a)(\forall v)(instancia(c,s) \wedge caracteristica(s,a,v) \rightarrow caracteristica(c,a,v))$$

LINGUAGENS LÓGICAS

- Programação declarativa
  - **O que se sabe** acerca do problema a resolver
  - **Inferência** a partir do que se sabe
- Representação de conhecimento
  - **Factos**
  - **Regras**
- Exemplo
  - ***PROLOG***

EXEMPLO

```
subclasse(ave, vertebrado).
subclasse(mamifero, vertebrado).
subclasse(aguia, ave).
subclasse(tigre, mamifero).
instancia(aquila, aguia).
caracteristica(ave, cobertura, penas).
caracteristica(aguia, cor, cinzenta).

instancia(I, S) :-
   subclasse(C, S),
   instancia(I, C).

caracteristica(X, A, V) :-
       instancia(X, S),
       caracteristica(S, A, V).
```

A REDE SEMÂNTICA (*Semantic Web*)

Resource Description Framework (*RDF*)

```
@prefix : <http://www.example.org/>.
:john    a           :Person.
:john    :hasMother  :susan.
:john    :hasFather  :richard.
:richard :hasBrother :luke.
```

N3

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
         xmlns:ns="http://www.example.org/#">
  <ns:Person rdf:about="http://www.example.org/#john">
    <ns:hasMother rdf:resource="http://www.example.org/#susan" />
    <ns:hasFather>
      <rdf:Description rdf:about="http://www.example.org/#richard">
        <ns:hasBrother rdf:resource="http://www.example.org/#luke" />
      </rdf:Description>
    </ns:hasFather>
  </ns:Person>
</rdf:RDF>
```

XML

```
@prefix : <http://www.example.org/> .
:richard :hasSister :rebecca
{ ?a :hasFather ?b . ?b :hasSister ?c . } => { ?a :hasAunt ?c } .
```

LÓGICA DE PRIMEIRA ORDEM

- A lógica é um **método fundamental para representação de conhecimento**, proporcionando uma base onde outros métodos podem ser suportados.
- Os **mecanismos de inferência** em lógica são **computacionalmente complexos**.
- É todavia possível definir **restrições**, nomeadamente à lógica de primeira ordem, com **poder representacional suficiente** para um determinado domínio, e **passíveis de um tratamento computacionalmente eficiente**.

LÓGICA DE PRIMEIRA ORDEM

VANTAGENS DE UTILIZAÇÃO:

- Representação declarativa
- Representação uniforme
- Poder expressivo
- Processos de inferência *correctos*
- Semântica bem definida

DESVANTAGENS DE UTILIZAÇÃO:

- Dificuldade de estruturação do conhecimento
- Dificuldade de representação de conhecimento procedimental
- Monotonicidade
- Semi-decidibilidade

CORRECÇÃO, COMPLETÚDE, DECIDIBILIDADE

Um sistema formal S é ***correcto*** se toda a fórmula p válida em relação aos axiomas de S é também um teorema de S

$$S \vdash p \rightarrow S \models p$$

Um sistema formal é ***completo*** se todos os seus teoremas são válidos

$$S \models p \rightarrow S \vdash p$$

Um sistema formal é ***decidível*** se existe um método que permita saber se uma determinada fórmula é ou não um teorema, de forma correcta e num número finito de passos

CORRECÇÃO, COMPLETÚDE, DECIDIBILIDADE

CORRECTO significa que, partindo das premissas, o mecanismo de inferência **nunca chegará a conclusões que não sejam consequência lógica** dessas premissas.

"*Tudo o que se prova é correcto*"

COMPLETO significa que o mecanismo de inferência é capaz de **gerar todas as consequências lógicas** possíveis do conjunto de premissas.

"*Até que ponto se pode provar tudo o que é verdadeiro*"
(dentro dos limites do cálculo)

DECIDÍVEL significa que **é sempre possível demonstrar** se uma determinada **fórmula é ou não consequência lógica** do conjunto de premissas.

LÓGICA DE PRIMEIRA ORDEM

A lógica de primeira ordem é ***completa***

A lógica de primeira ordem é ***correcta***

A lógica de primeira ordem é ***semi-decidível***

TEOREMAS DA INCOMPLETÚDE DE GÖDEL

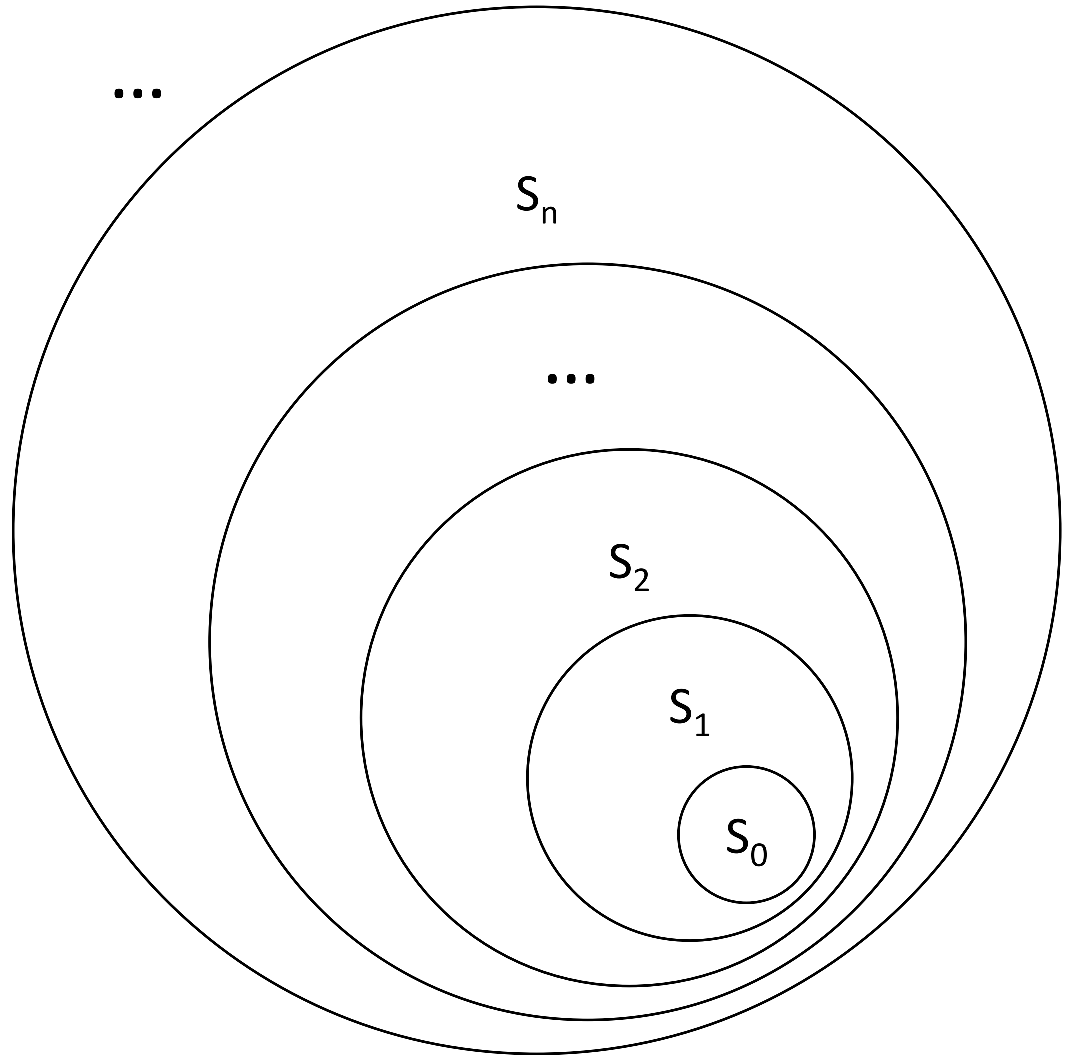
- **1º Teorema**
  - **Nenhum sistema consistente de axiomas** cujos teoremas podem ser enumerados por um "procedimento efectivo" (por exemplo, um **programa de computador** ou qualquer tipo de **algoritmo**) é capaz de provar todas as verdades sobre as relações dos números naturais
  - Para qualquer sistema desse tipo, haverá sempre afirmações sobre os números naturais que são verdadeiras, mas que **não são prováveis dentro do sistema**

- **2º Teorema**
  - **Um sistema não pode demonstrar a sua própria consistência**

LIMITES COGNITIVOS DE SISTEMAS SIMBÓLICOS

Os sistemas simbólicos são intrinsecamente limitados na capacidade de representação de conhecimento

Existirá sempre um âmbito infinito que não é passível de representação



REFERÊNCIAS

[Russel & Norvig, 2003]
S. Russell, P. Norvig, “Artificial Intelligence: A Modern Approach”, 2nd Ed., Prentice Hall, 2003

[Dean *et al*., 1995]
T. Dean, J. Allen, Y. Aloimonos, “Artificial Intelligence Theory and Practice”, Benjamin/Cummings, 1995.

[Jackson, 1990]
Peter Jackson, “Introduction to Expert Systems”, Addison-Wesley, 1990.

[Cawsey, 1998]
Alison Cawsey, “The Essence of Artificial Intelligence”, Prentice Hall, 1998.

[Rich & Knight, 1993]
E. Rich, K. Knight, “Artificial Intelligence”, McGraw-Hill, 1993.

[Winston, 1992]
P. Winston, “Artificial Intelligence”, Addison-Wesley, 1992.

[Reichgelt, 1991]
Han Reichgelt, “Knowledge Representation - An AI Perspective”, Ablex Publishing, 1991.

[Anderson, 1995]
James A. Anderson, “A Introduction to Neural Networks”, MIT Press, 1995.

[Kowalsky, 1979]
Robert Kowalsky, “Logic for Problem Solving”, North-Holland, 1979

[Gödel, 1962]
Kurt Gödel, “On Formally Undecidable Propositions of Principia Mathematica and
Related Systems”, Dover, 1962