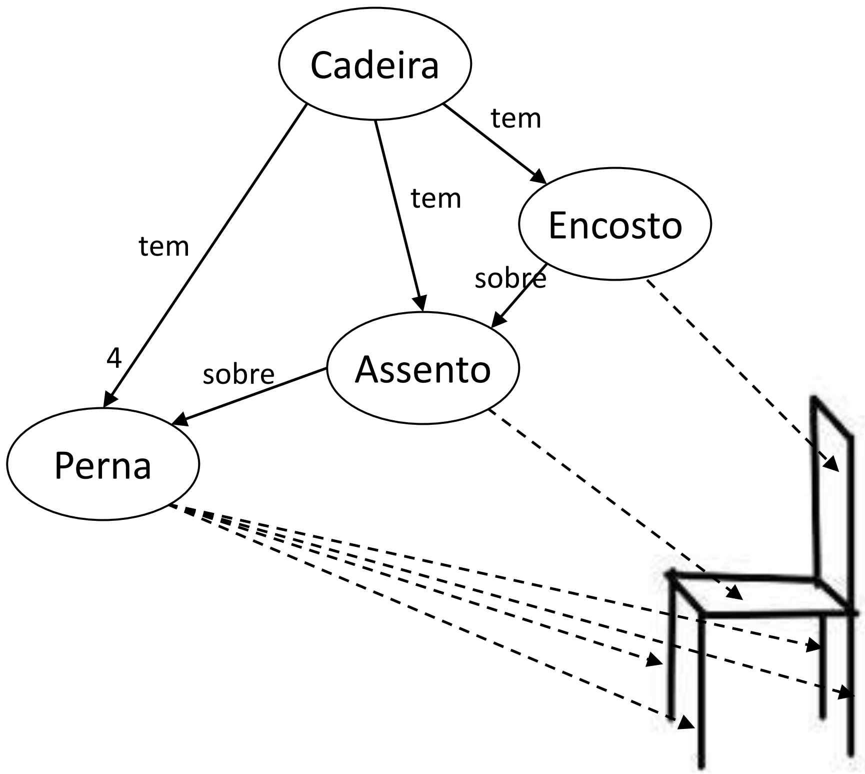
REPRESENTAÇÃO DE CONHECIMENTO

Luís Morgado

ISEL-DEETC

REPRESENTAÇÃO DE CONHECIMENTO

CONSTRUÇÃO DE SIGNIFICADO ATRAVÉS DE RELACIONAMENTO



Representação da realidade ≠ Realidade

REPRESENTAÇÃO

Aspectos de uma representação:

- A **notação** (***sintaxe***) a utilizar
- A **denotação** (***semântica***) das entidades representadas
- A forma de **manipulação** (***inferência***) das entidades representadas

REPRESENTAÇÃO = NOTAÇÃO +
DENOTAÇÃO +
MANIPULAÇÃO

REPRESENTAÇÃO

NOTAÇÃO

Normalmente designada por ***sintaxe***, corresponde à especificação das **convenções de forma** associadas a uma determinada representação

DENOTAÇÃO

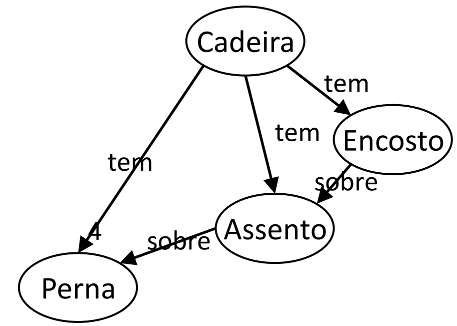
Normalmente designada por ***semântica***, corresponde à **atribuição de significado** às entidades especificadas na notação (conteúdo semântico)

MANIPULAÇÃO

Corresponde à especificação de um **modelo computacional** que define a **forma como os elementos da representação devem ser manipulados**, de acordo com as convenções semânticas (**inferência**, **raciocínio**)

REPRESENTAÇÃO

Semântica ***Sintaxe***



Factos

Mapeamento baseado na semântica

Factos representados formalmente

Raciocínio formal

Inferência

Conclusões

Mapeamento baseado na semântica

Conclusões representadas formalmente

TIPOS DE REPRESENTAÇÃO

DECLARATIVO (Denotacional)

- Representação **declarativa** de *o que* se sabe acerca de um domínio
- Controlo não representado explicitamente
- ***“Saber que …”***

PROCEDIMENTAL (Imperativo)

- Representação **procedimental** de *como* obter um resultado específico
- Controlo representado explicitamente
- ***“Saber como …”***

EXEMPLO: FACTORIAL

Especificação declarativa

```
factorial(0, 1).
factorial(N, Fact) :-
   N > 0,
   N1 is N - 1,
   factorial(N1, FactN1),
   Fact is N * FactN1.
```

Especificação procedimental

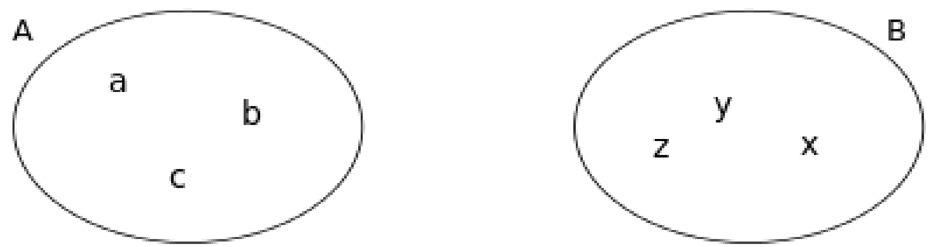
```
def factorial(n):
   if n == 0:
      return 1
   elif n > 0:
      fact = 1
      i = 1
      while i <= n:
         fact = fact * i
         i += 1
      return fact
```

REPRESENTAÇÃO DE CONHECIMENTO

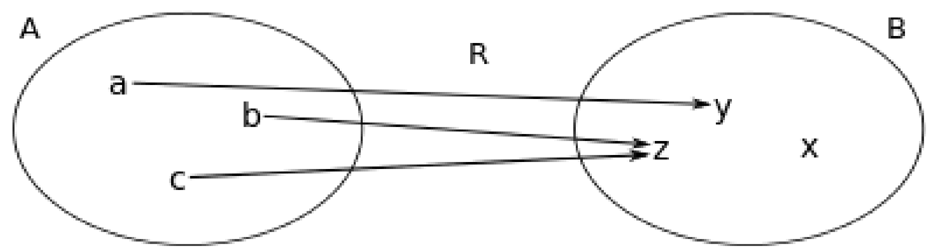
REPRESENTAÇÃO SOB A FORMA DE RELAÇÕES

RELAÇÃO

- Descrição das associações entre propriedades de objectos
- Uma relação $\boldsymbol{L}$ sobre os conjuntos $\boldsymbol{X_1, \ldots, X_k}$ é um subconjunto do seu produto cartesiano, i.e. $\boldsymbol{L \subseteq X_1 \times \ldots \times X_k}$



A x B = {(a,x), (b,x), (c,x), (a,y), (b,y), (c,y), (a,z), (b,z), (c,z)}



R = {(a,y), (b,z), (c,z)}

REPRESENTAÇÃO SOB A FORMA DE RELAÇÕES

EXEMPLO

LIVRO $\longrightarrow$ { TÍTULO $\rightarrow$ Artificial Intelligence: A Modern Approach
EDITORA $\rightarrow$ Prentice Hall

AUTOR $\longrightarrow$ { NOME $\rightarrow$ Stuart
APELIDO $\rightarrow$ Russell

AUTOR $\longrightarrow$ { NOME $\rightarrow$ Peter
APELIDO $\rightarrow$ Norvig

Esquema de relação:

AUTOR(NOME, APELIDO)

AUTOR = { (Stuart, Russel), (Peter Norvig) }

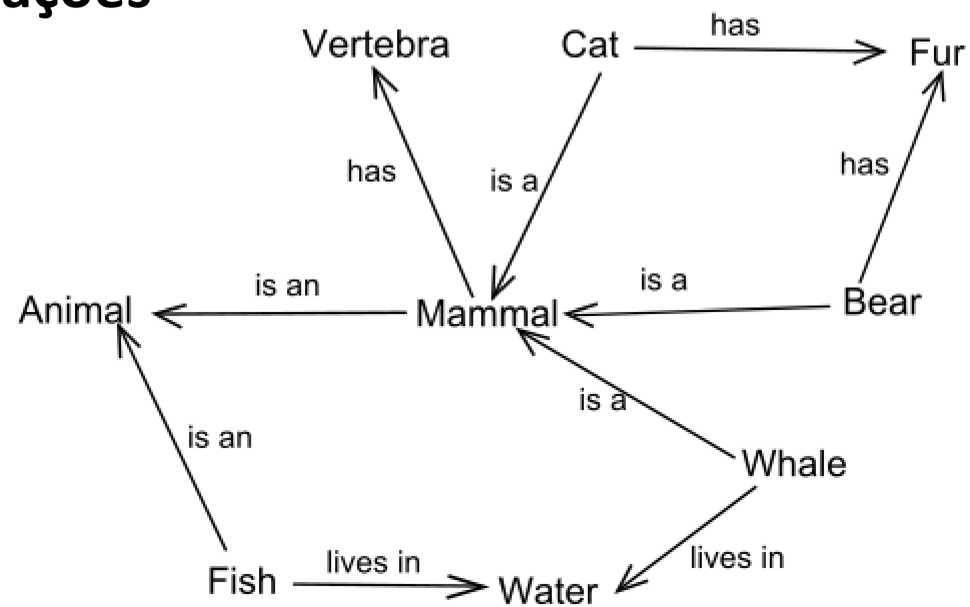
Tabela:

AUTOR	
NOME	**APELIDO**
Stuart	Russell
Peter	Norvig

ÁLGEBRA RELACIONAL $\rightarrow$ BASES DE DADOS RELACIONAIS

REDES SEMÂNTICAS

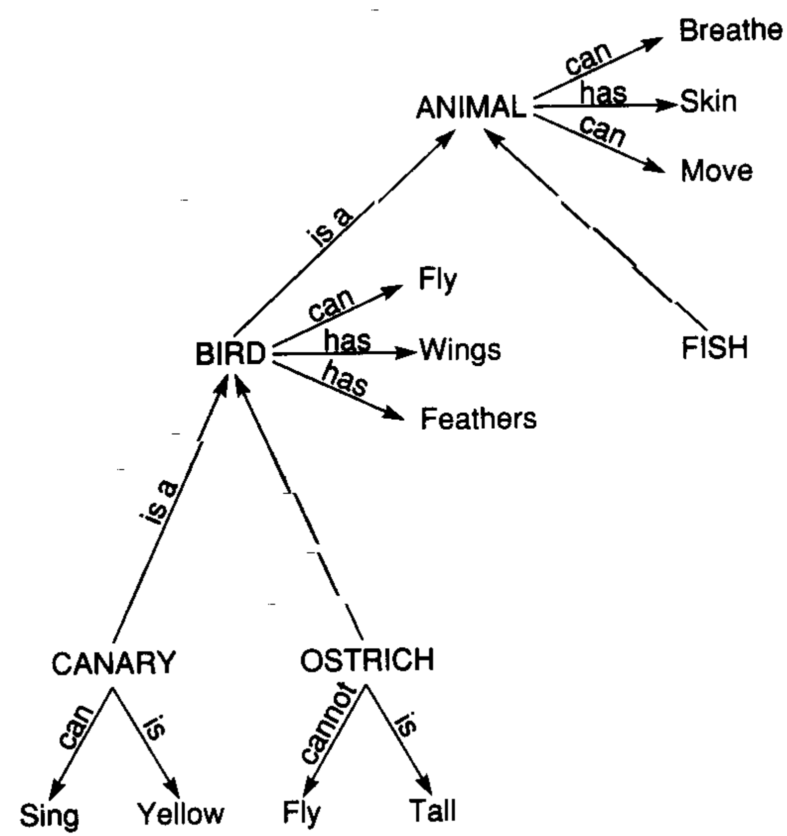
- O **significado de um conceito** resulta da forma **como está ligado** a outros conceitos
- Representação de conhecimento sob a forma de **grafos** (***redes semânticas***)
  - **Nós** representam **objectos** ou **conceitos**
  - **Arcos** representam **relações**
- Propostas por Ross Quillian (1963)



REDES SEMÂNTICAS

MANIPULAÇÃO

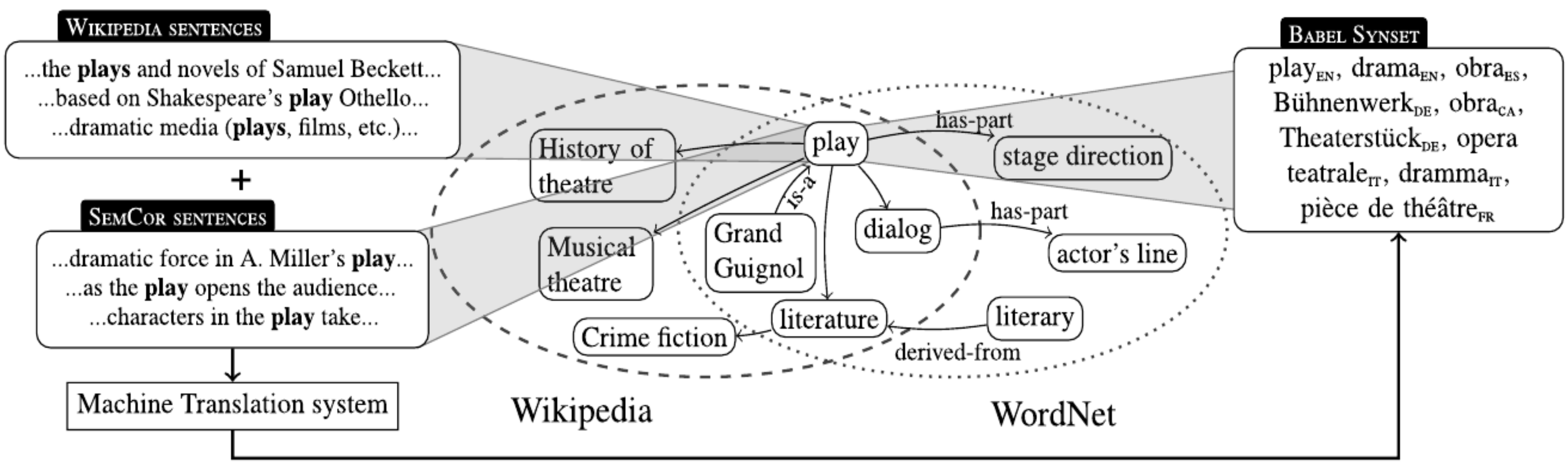
- O acesso a informação numa rede semântica é um problema de procura em grafos
- Mecanismo de **inferência**
  - Baseado em relações de **herança** (generalização)
    - **IS-A**
    - **INSTANCE-OF**
- Exemplo:
  - Uma ave **é** um animal logo **herda** as propriedades de um animal



REDES SEMÂNTICAS

EXEMPLO

- *BabelNet*
  - Rede semântica lexicográfica multilingue
  - *http://babelnet.org*



Ontologia

– Definição formal de tipos, propriedade e inter-relações de entidades que constituem um determinado domínio de discurso

REDES SEMÂNTICAS

PROBLEMAS

- **Inadequação lógica**
  - Significado impreciso de objectos, conceitos e relações
- **Inadequação de negação**
  - A representação da negação não é directamente suportada
- **Inadequação inferencial**
  - Obter uma informação específica pode ser muito ineficiente

REPRESENTAÇÃO DE CONHECIMENTO

REPRESENTAÇÃO EM LÓGICA FORMAL

“Logic studies the relationship of implication between assumptions and conclusions…”

In Logic for Problem Solving - Robert Kowalski

Relações representam **propriedades** que **atribuem valores de verdade** a *tuplos* de valores

Propriedades representadas por funções booleanas

$P: X \rightarrow \{\text{verdade, falso}\}$

Designadas **predicados**

LÓGICA FORMAL

- **Proposições**
  - São todas as expressões às quais é possível atribuir um valor de verdade ou falso
- **Princípio da não contradição**
  - Uma proposição não pode ser simultaneamente verdadeira e falsa
- **Princípio do terceiro excluído**
  - Uma proposição ou é verdadeira ou é falsa (não existe uma terceira opção)
- **Proposições Equivalentes**
  - Têm o mesmo valor lógico

LÓGICA PROPOSICIONAL

SINTAXE

Símbolos representam:

- **Proposições**
- **Conectivas lógicas**
  - Definem formas de combinar proposições

Por exemplo:

c - “Chove”
g - “Utilizar o guarda chuva”

$c \rightarrow g$: “Se chove então utilizar o guarda chuva”

LÓGICA PROPOSICIONAL

CONECTIVAS LÓGICAS

- Negação (não): $\neg$
- Conjunção (e): $\wedge$
- Disjunção (ou): $\vee$
- Implicação (se...então): $\rightarrow$
- Equivalência (se e apenas se): $\leftrightarrow$

LÓGICA PROPOSICIONAL

INFERÊNCIA

As regras de inferência permitem obter conclusões a partir dos axiomas do domínio. Uma **regra de inferência** é composta por duas partes:

- ***Condições***
- ***Conclusão***

Exemplo:

modus ponens

$$\frac{\begin{array}{c} A \\ A \rightarrow B \end{array}}{B}$$

princípio da resolução

$$\frac{\begin{array}{c} A \vee B \\ \neg A \vee C \end{array}}{B \vee C}$$

EXEMPLO

Um especialista de ar condicionado afirma que quando o sistema de refrigeração está a funcionar correctamente a sua parte exterior deve estar fria.

Verificou-se que a parte exterior está fria.

Símbolos de proposição :

$sistema\_ok$: “O sistema funciona correctamente”
$exterior\_frio$: “A parte exterior está fria”

Conhecimento do domínio:

$$exterior\_frio$$

$$exterior\_frio \rightarrow sistema\_ok$$

EXEMPLO

Um especialista de ar condicionado afirma que quando o sistema de refrigeração está a funcionar correctamente a sua parte exterior deve estar fria.

Verificou-se que a parte exterior está fria.

Através da regra de inferência *modus ponens* podemos concluir que o sistema funciona correctamente:

$$\frac{\begin{array}{l} exterior\_frio \\ exterior\_frio \rightarrow sistema\_ok \end{array}}{sistema\_ok}$$

Com base nas regras de **inferência** (em particular no *princípio da resolução*) é possível realizar **sistemas automáticos de dedução**, apesar de, em alguns casos, esse processo poder ser computacionalmente exigente.

LÓGICA PROPOSICIONAL

No cálculo proposicional é possível definir declarações como:

"*se o João é homem, então o João é mortal*":

$$a \equiv "O\ João\ é\ homem"$$

$$b \equiv "O\ João\ é\ mortal"$$

$$(a \rightarrow b)$$

O **problema deste tipo de representação** reside na **ausência de estruturação das fórmulas** e na **impossibilidade de generalização**.

No exemplo anterior, o predicado homem designa uma propriedade do seu argumento.

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 de 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***

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***

LINGUAGEM PROLOG

EXEMPLO

- Representação e processamento de informação genealógica

Factos conhecidos:

- O João é **pai** da Rita
- O João é **pai** do Rui
- A Rita é **mãe** do Carlos
- A Rita é **mãe** da Joana
- O Rui é **pai** do José
- O José é **pai** do António
- O Carlos é **pai** do Joaquim

Especificação em Prolog:

```
pai(joao, rita).
pai(joao, rui).
mae(rita, carlos).
mae(rita, joana).
pai(rui, jose).
pai(jose, antonio).
pai(carlos, joaquim).
```

LINGUAGEM PROLOG

EXEMPLO

Regras:

```
filho(X, Y) :- pai(Y, X).
filho(X, Y) :- mae(Y, X).

avo(X, Y) :-
  filho(Z, X),
  filho(Y, Z).

irmao(X, Y) :-
  filho(X, Z),
  filho(Y, Z),
  X \= Y.

tio(X, Y) :-
  filho(Y, Z),
  irmao(X, Z).
```

Interrogação:

```
?- tio(joao, X).
X = carlos
```

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

$$S \vdash p \rightarrow S \vDash p$$

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

$$S \vDash p \rightarrow S \vdash p$$

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***

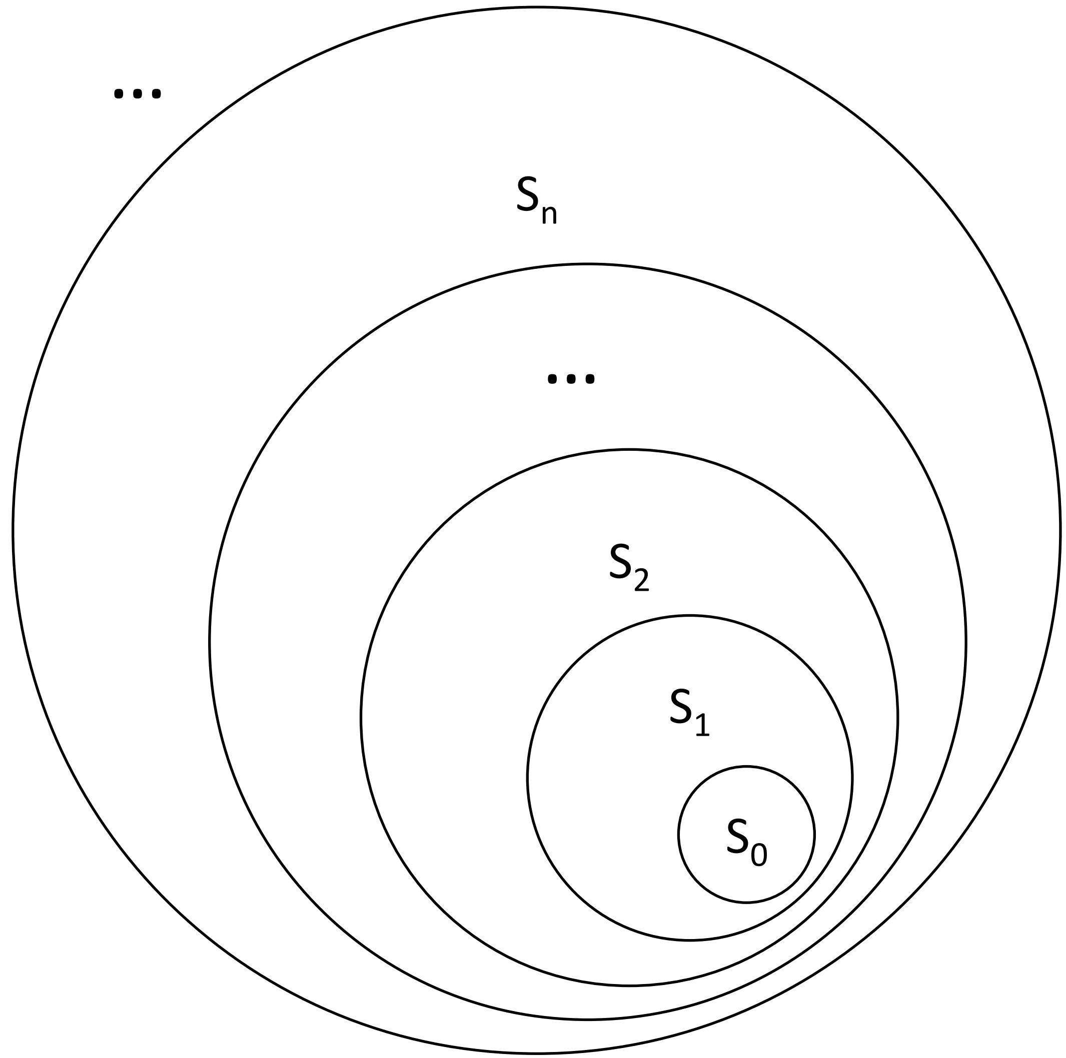
TEOREMAS DA INCOMPLETÚDE DE GÖDEL

- **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