

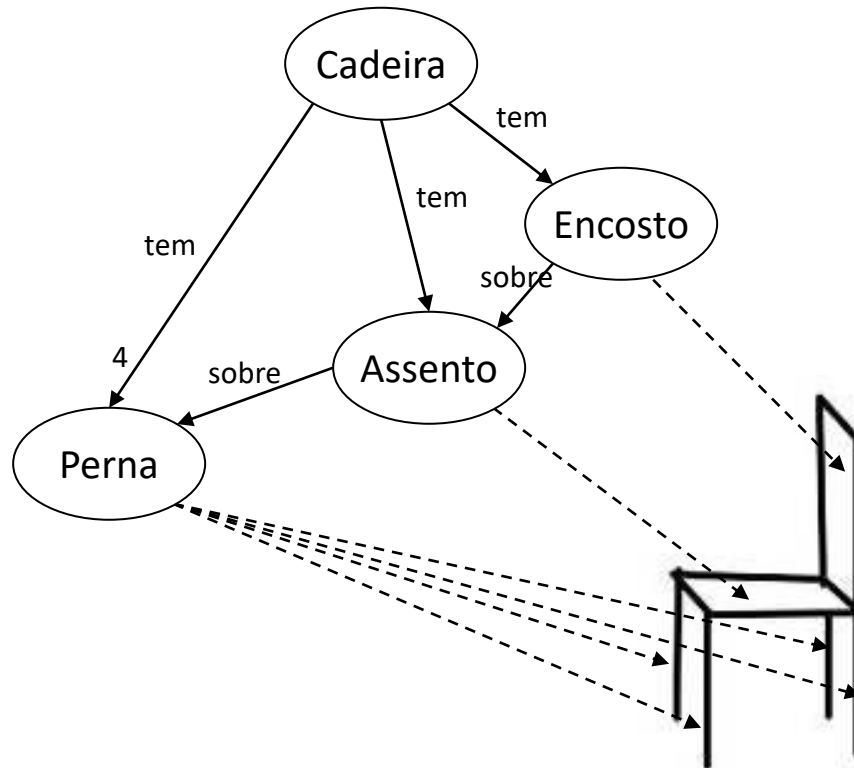
REPRESENTAÇÃO DE CONHECIMENTO

Luís Morgado

ISEL-ADEETC

REPRESENTAÇÃO DE CONHECIMENTO

CONSTRUÇÃO DE SIGNIFICADO ATRAVÉS DE RELACIONAMENTO



Representação da realidade \neq 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

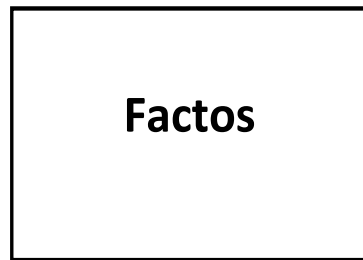
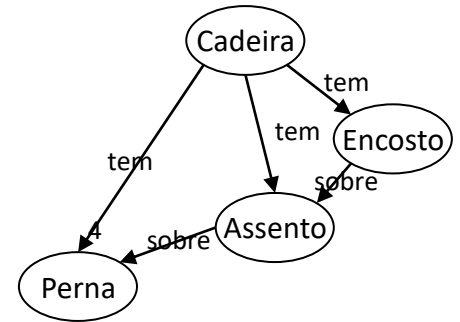
Corresponde à especificação de um **modelo computacional** que define a **forma como os elementos da representação devem ser manipulados**, de acordo com as convenções semânticas (**inferência, raciocínio**)

REPRESENTAÇÃO



Semântica

Sintaxe

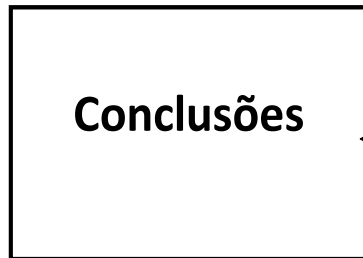


Mapeamento
baseado na
semântica

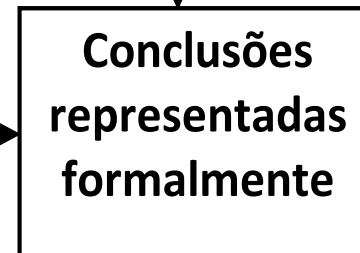


Raciocínio
formal

Inferência



Mapeamento
baseado na
semântica



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

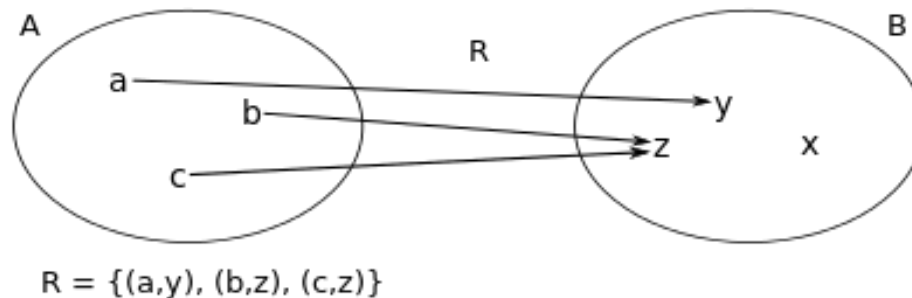
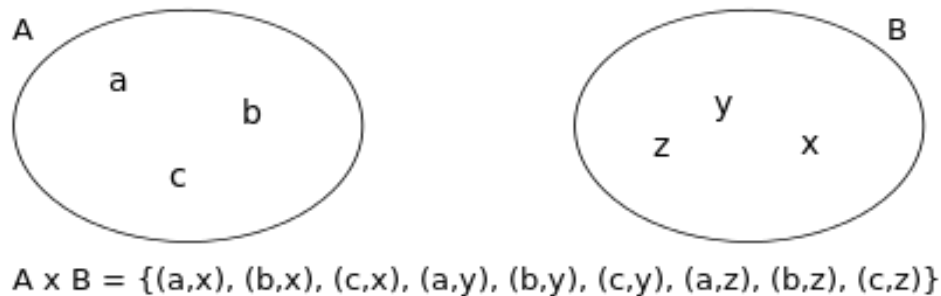
```
def factorial(n):  
    if n == 0:  
        return 1  
    elif n > 0:  
        fact = 1  
        i = 1  
        while i <= n:  
            fact = fact * i  
            i += 1  
    return fact
```

REPRESENTAÇÃO DE CONHECIMENTO

REPRESENTAÇÃO SOB A FORMA DE RELAÇÕES

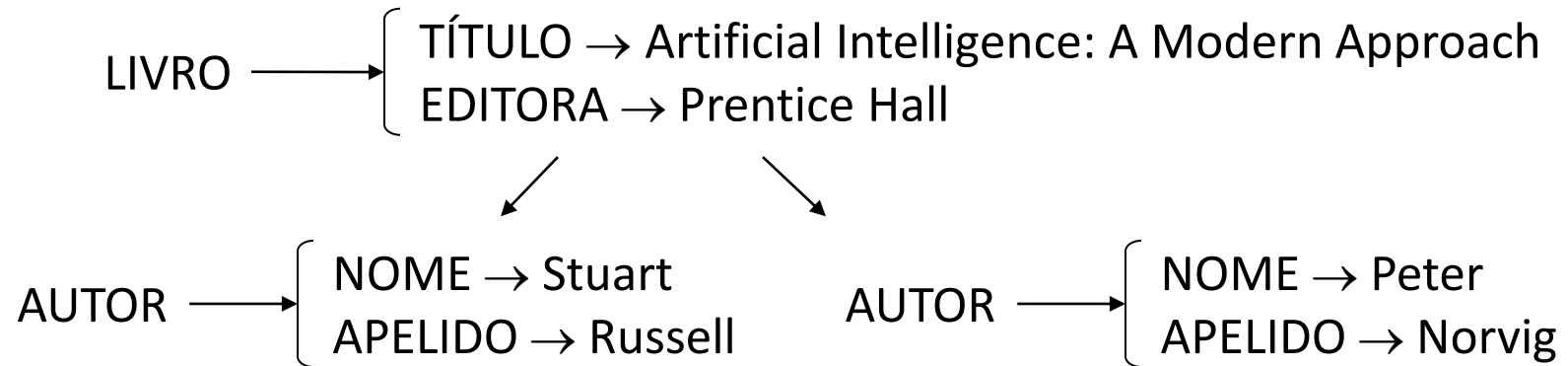
RELAÇÃO

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



REPRESENTAÇÃO SOB A FORMA DE RELAÇÕES

EXEMPLO



Esquema de relação:

AUTOR(NOME, APELIDO)

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

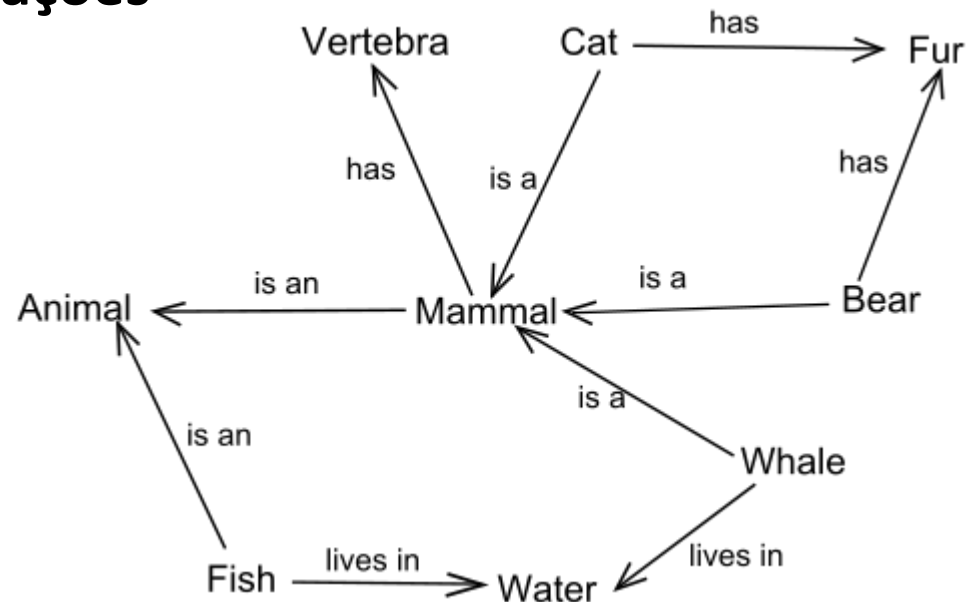
Tabela:

AUTOR	
NOME	APELIDO
Stuart	Russell
Peter	Norvig

ÁLGEBRA RELACIONAL → BASES DE DADOS RELACIONAIS

REDES SEMÂNTICAS

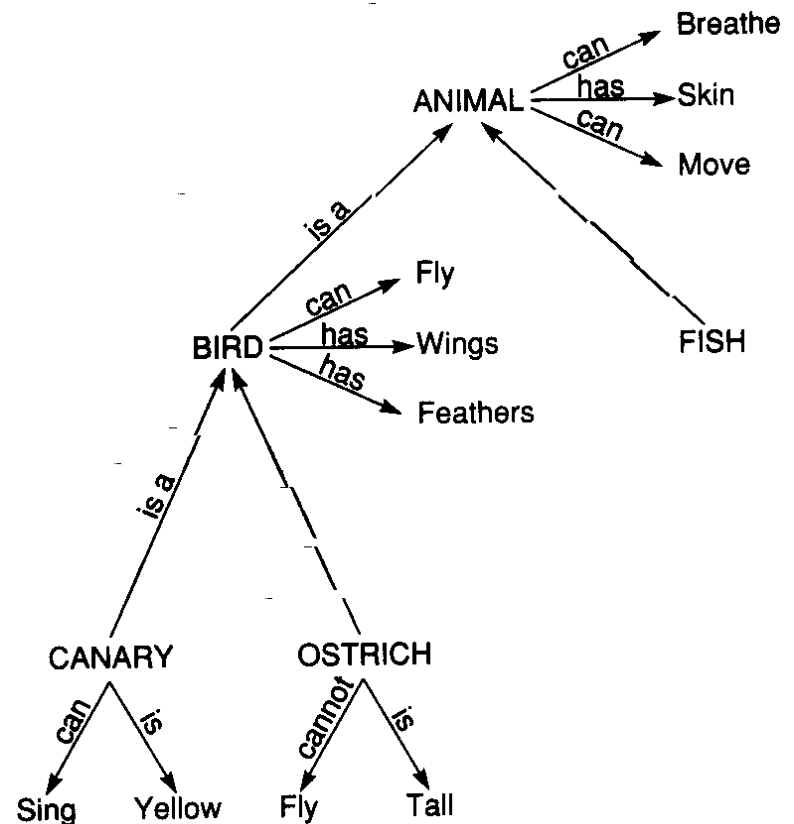
- O significado de um conceito resulta da forma como está ligado a outros conceitos
- Representação de conhecimento sob a forma de grafos (*redes semânticas*)
 - Nós representam **objectos** ou **conceitos**
 - Arcos representam **relações**
- Propostas por Ross Quillian (1963)



REDES SEMÂNTICAS

MANIPULAÇÃO

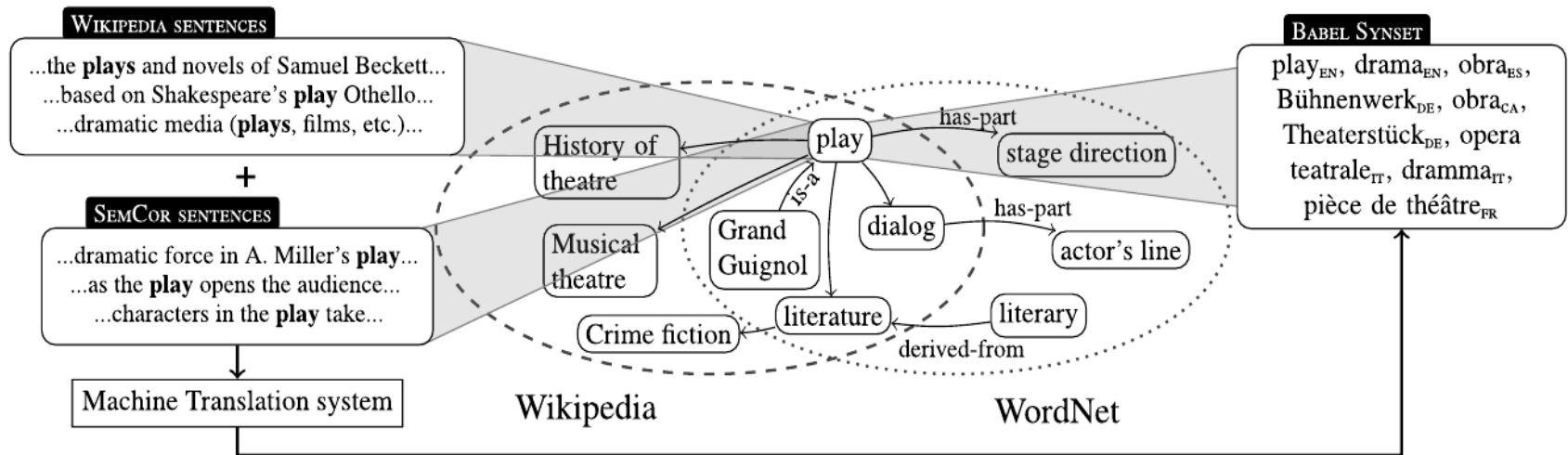
- O acesso a informação numa rede semântica é um problema de procura em grafos
- Mecanismo de **inferência**
 - Baseado em relações de **herança** (generalização)
 - IS-A
 - INSTANCE-OF
- Exemplo:
 - Uma ave é um animal logo **herda** as propriedades de um animal



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

$$\frac{A \quad A \rightarrow B}{B}$$

princípio da resolução

$$\frac{A \vee B \quad \neg A \vee C}{B \vee C}$$

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:

$$\frac{\textit{exterior_frio} \quad \textit{exterior_frio} \rightarrow \textit{sistema_ok}}{\textit{sistema_ok}}$$

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.

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