

# Sistemas Baseados em Conhecimento

## Aula 13

Renata Wassermann

`renata@ime.usp.br`

2019

# Ontologias

Definição (Gruber, 1992)

*“Especificação explícita de uma conceitualização”*

# Ontologias

## Definição (Gruber, 1992)

*“Especificação explícita de uma conceitualização”*

- Usadas para representação do conhecimento **reuso** e **compartilhamento**
- **Conceitos** e **relações** que descrevem o domínio
- Dependem da aplicação

# Para que ontologias?

- Dados de múltiplas fontes (ex.: histórico de paciente, exames de laboratório, exame clínico).
- Como encontrar dados?
- Se encontrar, como interpretar dados armazenados? (ex.: medicações em uso)
- Como integrar e usar os dados?

# Recuperação

Tumor de mama (20)

Neoplasia maligna de mama (25)

Carcinoma de mama (25)

- Anotações sobre os três conceitos acima.

# Recuperação

Tumor de mama (20)

Neoplasia maligna de mama (25)

Carcinoma de mama (25)

- Anotações sobre os três conceitos acima.
- Sem ontologia, busca por “tumor de mama” devolve 20 anotações.

# Recuperação

Tumor de mama (20)

Neoplasia maligna de mama (25)

Carcinoma de mama (25)

- Anotações sobre os três conceitos acima.
- Sem ontologia, busca por “tumor de mama” devolve 20 anotações.
- Com ontologia, busca devolve 70 anotações.

# Integração

Se cada grupo usa seu modelo, como integrar informação?

- Nomes iguais para conceitos diferentes  
ex.: Paciente (interno ou qualquer)
- Nomes diferentes para conceitos iguais  
ex.: Tumor e Neoplasia



# Tipos de conhecimento

O paciente é adulto.

O paciente tem tumor ou não tem tumor.

Se o paciente é solteiro então não é casado.

# Tipos de conhecimento

O paciente é adulto. **sintético**

O paciente tem tumor ou não tem tumor.

Se o paciente é solteiro então não é casado.

# Tipos de conhecimento

O paciente é adulto. **sintético**

O paciente tem tumor ou não tem tumor. **analítico - lógico**

Se o paciente é solteiro então não é casado.

# Tipos de conhecimento

O paciente é adulto. **sintético**

O paciente tem tumor ou não tem tumor. **analítico - lógico**

Se o paciente é solteiro então não é casado.

**analítico - terminológico**

# Ontologias - conhecimento terminológico

- Conceitos (ou classes)
- Propriedades (ou papéis ou slots)
- Restrições (de papéis ou facets)

Decisões: Red apple/Apple red

**Importante:** Análise ontológica deve vir antes de qualquer representação.

# Semântica em Lógica

- Interpretação = Domínio ( $D$ ) + Mapeamento ( $I$ )
- Como inferir  $\text{mammal}(\text{Lulu})$  de  $\text{dog}(\text{Lulu})$ ?
- $D = \{d\}$   
 $I(\text{Lulu}) = d$   
 $I(\text{mammal}) = \{\}$   
 $I(\text{dog}) = \{d\}$ .
- De acordo com esta interpretação, Lulu é o nome do objeto  $d$ , que é um cachorro e não é mamífero.

# Evitando Interpretações Indesejadas

- Queremos garantir a relação **semântica**  
 $I(dog) \subseteq I(mammal)$
- Fazemos isso adicionando fórmulas (**objetos sintáticos**):  
 $\forall x(dog(x) \rightarrow mammal(x))$

# O que são lógicas de descrição? (Horrocks)

- Família de formalismos para representação de conhecimento baseados em lógica adequados para a representação e raciocínio sobre conhecimento terminológico
- Descendentes das redes semânticas, frames, KL-ONE...
- ... também conhecidos como sistemas terminológicos, linguagens de conceitos, etc.



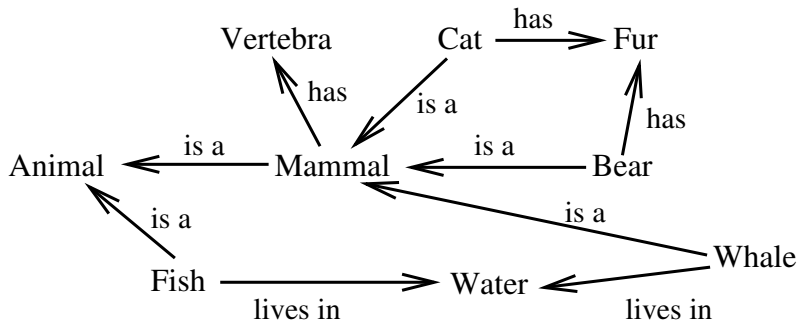
# Redes Semânticas

- Peirce 1909: “Lógica do futuro”
- Tese de Quillian (anos 60): “Semantic memory”
- O significado de um conceito vem da relação com outros conceitos.
- A informação é guardada em nós interconectados por arcos rotulados.

Cat  $\xrightarrow{\text{is a}}$  Mammal

Fish  $\xrightarrow{\text{lives in}}$  Water

# Redes Semânticas



Woods “What’s in a link?” (1975)

cat  $\xrightarrow{\text{colour}}$  black

Woods "What's in a link?" (1975)

cat  $\xrightarrow{\text{colour}}$  black

- Todos os gatos são inteiramente pretos.

Woods “What’s in a link?” (1975)

cat  $\xrightarrow{\text{colour}}$  black

- Todos os gatos são inteiramente pretos.
- Todos os gatos têm algo de preto.

Woods “What’s in a link?” (1975)

cat  $\xrightarrow{\text{colour}}$  black

- Todos os gatos são inteiramente pretos.
- Todos os gatos têm algo de preto.
- Existe um gato inteiramente preto.

Woods “What’s in a link?” (1975)

cat  $\xrightarrow{\text{colour}}$  black

- Todos os gatos são inteiramente pretos.
- Todos os gatos têm algo de preto.
- Existe um gato inteiramente preto.
- Existe um gato que tem algo de preto.

# KL-ONE

- Tese do Brachman, Harvard, 1977
- William A. Woods, James G. Schmolze, The KL-ONE family, 1992
- Usado por empresas e institutos de pesquisa.
- Foco em PLN.
- Semântica clara.
- Primitivas para estruturar conceitos.
- Base para o que vem a seguir...



# KL-ONE

- Separação entre conhecimento terminológico e assertivo
- Significado externo de “is a” (conjuntos e instâncias)
- Herança múltipla
- Permite classificação automática

# KL-ONE

- Separação entre conhecimento terminológico e assertivo
- Significado externo de “is a” (conjuntos e instâncias)
- Herança múltipla
- Permite classificação automática  
Exemplo: inferir que “*Woman with sons*” é mais específico que “*Woman with children*”

## O começo da era moderna

“The Tractability of Subsumption in Frame-Based Description Languages”, Brachman and Levesque, 1984.

*Here we present evidence as to how the cost of computing one kind of inference is directly related to the expressiveness of the representation language. As it turns out, this cost is perilously sensitive to small changes in the representation language. Even a seemingly simple frame-based description language can pose intractable computational obstacles.*

# Base de conhecimento em LD

- Descrição dos conceitos e suas propriedades.
- Descrição de situações concretas.

# Conceitos

- Representam classes (conjuntos)
- Podem ser atômicos ou construídos:
  - Female
  - Female  $\sqcap$  Human

# Papéis (Roles)

- Representam relações (propriedades)
- Podem ser utilizados para restringir conceitos:
  - $\text{Female} \sqcap \text{Human} \sqcap \exists \text{haschild.Female}$
  - $\text{Female} \sqcap \forall \text{haschild.Human}$

# TBox

A TBox contém o conhecimento terminológico:

- Definições de conceitos (introduzem nomes para conceitos):  
 $\text{Father} \equiv \text{Man} \sqcap \exists \text{haschild}.\top$
- Axiomas (restringem os modelos):  
 $\text{Mother} \sqsubseteq \text{Woman}$   
 $\text{BlackCat} \sqsubseteq \text{Cat} \sqcap \forall \text{hascolour}.\text{Black}$

## Exemplo

Woman	$\equiv$	$\text{Person} \sqcap \text{Female}$
Man	$\equiv$	$\text{Person} \sqcap \neg \text{Woman}$
Mother	$\equiv$	$\text{Woman} \sqcap \exists \text{hasChild}.\text{Person}$
Father	$\equiv$	$\text{Man} \sqcap \exists \text{hasChild}.\text{Person}$
Parent	$\equiv$	$\text{Father} \sqcup \text{Mother}$
Grandmother	$\equiv$	$\text{Mother} \sqcap \exists \text{hasChild}.\text{Parent}$
MotherWithoutDaughter	$\equiv$	$\text{Mother} \sqcap \forall \text{hasChild}.\neg \text{Woman}$
Wife	$\equiv$	$\text{Woman} \sqcap \exists \text{hasHusband}.\text{Man}$



# ABox

A ABox contém asserções sobre indivíduos:

- Asserções de conceito:  
BlackCat(mimi)  
Mother  $\sqcap \exists \text{haschild.Woman}(\text{mary})$
- Asserções de papéis:  
haschild(mary,betty)

# A linguagem típica ( $\mathcal{ALC}$ )

- Attributive Concept Language with Complements (Schmidt-Schauß and Smolka, 1991).
- Construção de conceitos fechada sob operadores booleanos.
- Base para linguagens mais expressivas.

# Conceitos

$$C, D \longrightarrow \begin{array}{l} A \mid \\ \top \mid \\ \perp \mid \\ \neg C \mid \\ C \sqcap D \mid \\ C \sqcup D \mid \\ \forall R.C \mid \\ \exists R.C \end{array}$$

# Interpretações

$\mathcal{I}$ :

- $\Delta^{\mathcal{I}}$  (domínio da interpretação)
- função que atribui para cada:
  - $A$ , um conjunto  $A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$
  - $R$ , uma relação binária  $R^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$ .
  - $a$ , um elemento  $a^{\mathcal{I}} \in \Delta^{\mathcal{I}}$

## Estendendo a função de interpretação

$$\begin{aligned}\top^{\mathcal{I}} &= \Delta^{\mathcal{I}} \\ \perp^{\mathcal{I}} &= \emptyset \\ (\neg C)^{\mathcal{I}} &= \Delta^{\mathcal{I}} \setminus C^{\mathcal{I}} \\ (C \sqcap D)^{\mathcal{I}} &= C^{\mathcal{I}} \cap D^{\mathcal{I}} \\ (C \sqcup D)^{\mathcal{I}} &= C^{\mathcal{I}} \cup D^{\mathcal{I}} \\ (\exists R.C)^{\mathcal{I}} &= \{a \in \Delta^{\mathcal{I}} \mid \exists (a, b) \in R^{\mathcal{I}} \text{ e } b \in C^{\mathcal{I}}\} \\ (\forall R.C)^{\mathcal{I}} &= \{a \in \Delta^{\mathcal{I}} \mid \forall (a, b) \in R^{\mathcal{I}}, b \in C^{\mathcal{I}}\}\end{aligned}$$

# Semântica da TBox

Uma interpretação  $\mathcal{I}$  satisfaz

- $C \equiv D$  sse  $C^{\mathcal{I}} = D^{\mathcal{I}}$ .

# Semântica da TBox

Uma interpretação  $\mathcal{I}$  satisfaz

- $C \equiv D$  sse  $C^{\mathcal{I}} = D^{\mathcal{I}}$ .
- $C \sqsubseteq D$  sse  $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$ .

# Semântica da TBox

Uma interpretação  $\mathcal{I}$  satisfaz

- $C \equiv D$  sse  $C^{\mathcal{I}} = D^{\mathcal{I}}$ .
- $C \sqsubseteq D$  sse  $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$ .
- a TBox  $\mathcal{T}$  sse satisfaz todos os elementos de  $\mathcal{T}$



# Semântica da ABox

Uma interpretação  $\mathcal{I}$  satisfaz

- $C(a)$  sse  $a^{\mathcal{I}} \in C^{\mathcal{I}}$ .

# Semântica da ABox

Uma interpretação  $\mathcal{I}$  satisfaz

- $C(a)$  sse  $a^{\mathcal{I}} \in C^{\mathcal{I}}$ .
- $r(a, b)$  sse  $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in r^{\mathcal{I}}$ .

# Semântica da ABox

Uma interpretação  $\mathcal{I}$  satisfaz

- $C(a)$  sse  $a^{\mathcal{I}} \in C^{\mathcal{I}}$ .
- $r(a, b)$  sse  $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in r^{\mathcal{I}}$ .
- a ABox  $\mathcal{A}$  sse satisfaz todos os elementos de  $\mathcal{A}$

# Base de conhecimento $\mathcal{ALC}$

Uma Base de conhecimento  $\mathcal{ALC}$  é um par  $\Sigma = \langle \mathcal{T}, \mathcal{A} \rangle$ , onde

- $\mathcal{T}$  é uma TBox
- $\mathcal{A}$  é uma ABox

Uma interpretação  $\mathcal{I}$  é um *modelo* de  $\Sigma$  se satisfaz  $\mathcal{T}$  e  $\mathcal{A}$ .

Uma base de conhecimento  $\Sigma$  é *satisfatível* se admite um modelo.

# Consequência Lógica

$\Sigma \models \varphi$  sse todo modelo de  $\Sigma$  é um modelo de  $\varphi$

# Consequência Lógica

$\Sigma \models \varphi$  sse todo modelo de  $\Sigma$  é um modelo de  $\varphi$

$\exists \text{teaches.Course} \sqsubseteq \text{GraduateStudent} \sqcup \text{Professor}$

$\text{teaches}(\text{john}, \text{cs101})$

$\text{Course}(\text{cs101})$

$\neg \text{Professor}(\text{john})$

# Consequência Lógica

$\Sigma \models \varphi$  sse todo modelo de  $\Sigma$  é um modelo de  $\varphi$

$\exists \text{teaches.Course} \sqsubseteq \text{GraduateStudent} \sqcup \text{Professor}$

$\text{teaches}(\text{john}, \text{cs101})$

$\text{Course}(\text{cs101})$

$\neg \text{Professor}(\text{john})$

$\Sigma \models \text{GraduateStudent}(\text{john})$

# Tradução para LPO

Conceitos são traduzidos para fórmulas com uma variável livre:

$$t_x(A) = A(x)$$

$$t_x(C \sqcap D) = t_x(C) \wedge t_x(D)$$

$$t_x(\forall r.C) = \forall y(r(x, y) \rightarrow t_y(C))$$

$$t_x(\exists r.C) = \exists y(r(x, y) \wedge t_y(C))$$



# Tradução para LPO

Conceitos são traduzidos para fórmulas com uma variável livre:

$$t_x(A) = A(x)$$

$$t_x(C \sqcap D) = t_x(C) \wedge t_x(D)$$

$$t_x(\forall r.C) = \forall y(r(x, y) \rightarrow t_y(C))$$

$$t_x(\exists r.C) = \exists y(r(x, y) \wedge t_y(C))$$

Axiomas  $C \sqsubseteq D$  correspondem a

$$\forall x(t_x(C) \rightarrow t_x(D))$$

# Tradução para LPO

$\text{Cat} \sqsubseteq \text{Mammal}$

$\forall x(\text{Cat}(x) \rightarrow \text{Mammal}(x))$

# Tradução para LPO

$\text{Cat} \sqsubseteq \text{Mammal}$

$\forall x(\text{Cat}(x) \rightarrow \text{Mammal}(x))$

$\text{Student} \sqsubseteq \forall \text{hasPet} . (\text{Owl} \sqcup \text{Cat} \sqcup \text{Toad})$

$\forall x(\text{Student}(x) \rightarrow$   
 $\quad \forall y (\text{hasPet}(x,y) \rightarrow (\text{Owl}(y) \vee \text{Cat}(y) \vee \text{Toad}(y))))$

## Exemplo (Horrocks) - TBox

HogwartsStudent  $\equiv$  Student  $\sqcap$   $\exists$ attendsSchool.Hogwarts

HogwartsStudent  $\sqsubseteq$   $\forall$ hasPet.(Owl  $\sqcup$  Cat  $\sqcup$  Toad)

$\exists$ hasPet.Phoenix  $\sqsubseteq$  Wizard

Phoenix  $\sqsubseteq$   $\neg$ (Owl  $\sqcup$  Cat  $\sqcup$  Toad)

# Exemplo (Horrocks) - ABox

**Fatos:**

HogwartsStudent(harryPotter)

**Inferências:**

## Exemplo (Horrocks) - ABox

### Fatos:

HogwartsStudent(harryPotter)

### Inferências:

$\exists$ attendsSchool.Hogwarts(harryPotter)

## Exemplo (Horrocks) - ABox

### Fatos:

HogwartsStudent(harryPotter)  
hasPet(harryPotter, hedwig)

### Inferências:

$\exists$ attendsSchool.Hogwarts(harryPotter)

## Exemplo (Horrocks) - ABox

### Fatos:

HogwartsStudent(harryPotter)  
hasPet(harryPotter, hedwig)

### Inferências:

$\exists$ attendsSchool.Hogwarts(harryPotter)  
(Owl  $\sqcup$  Cat  $\sqcup$  Toad)(hedwig)



## Exemplo (Horrocks) - ABox

### Fatos:

HogwartsStudent(harryPotter)  
hasPet(harryPotter, hedwig)  
Phoenix(fawks)

### Inferências:

$\exists$ attendsSchool.Hogwarts(harryPotter)  
(Owl  $\sqcup$  Cat  $\sqcup$  Toad)(hedwig)

## Exemplo (Horrocks) - ABox

### Fatos:

HogwartsStudent(harryPotter)  
hasPet(harryPotter, hedwig)  
Phoenix(fawks)

### Inferências:

$\exists$ attendsSchool.Hogwarts(harryPotter)  
(Owl  $\sqcup$  Cat  $\sqcup$  Toad)(hedwig)  
 $\neg$ hasPet(harryPotter,fawks)

## Exemplo (Horrocks) - ABox

### Fatos:

HogwartsStudent(harryPotter)  
hasPet(harryPotter, hedwig)  
Phoenix(fawks)  
hasPet(dumbledore, fawks)

### Inferências:

$\exists$  attendsSchool.Hogwarts(harryPotter)  
(Owl  $\sqcup$  Cat  $\sqcup$  Toad)(hedwig)  
 $\neg$  hasPet(harryPotter, fawks)

## Exemplo (Horrocks) - ABox

### Fatos:

HogwartsStudent(harryPotter)  
hasPet(harryPotter, hedwig)  
Phoenix(fawks)  
hasPet(dumbledore, fawks)

### Inferências:

$\exists$  attendsSchool.Hogwarts(harryPotter)  
(Owl  $\sqcup$  Cat  $\sqcup$  Toad)(hedwig)  
 $\neg$  hasPet(harryPotter, fawks)  
Wizard(dumbledore) and  
 $\neg$  HogwartsStudent(dumbledore)

# Ontologias vs. Banco de Dados

**Banco de Dados:**

**Ontologias:**

# Ontologias vs. Banco de Dados

## Banco de Dados:

- Mundo fechado  
(não está = falso)

## Ontologias:

- Mundo aberto  
(não está = desconhecido)

# Ontologias vs. Banco de Dados

## Banco de Dados:

- Mundo fechado  
(não está = falso)
- UNA  
(cada indivíduo tem nome único)

## Ontologias:

- Mundo aberto  
(não está = desconhecido)
- Não tem UNA  
(indivíduo pode ter vários nomes)

# Ontologias vs. Banco de Dados

## Banco de Dados:

- Mundo fechado  
(não está = falso)
- UNA  
(cada indivíduo tem nome único)
- Esquema restringe dados

## Ontologias:

- Mundo aberto  
(não está = desconhecido)
- Não tem UNA  
(indivíduo pode ter vários nomes)
- Axiomas usados para inferência



## Exemplo - continuação

Dados os fatos:

`hasFriend(harryPotter,ronWeasley)`

`hasFriend(harryPotter, hermioneGranger)`

## Exemplo - continuação

Dados os fatos:

`hasFriend(harryPotter,ronWeasley)`

`hasFriend(harryPotter, hermioneGranger)`

**Pergunta:** Draco é amigo de Harry Potter?

## Exemplo - continuação

Dados os fatos:

`hasFriend(harryPotter,ronWeasley)`

`hasFriend(harryPotter, hermioneGranger)`

**Pergunta:** Draco é amigo de Harry Potter?

BD: não.

Ontologia: não sei (mundo aberto).

## Exemplo - continuação

Pergunta: Quantos amigos Harry Potter tem?

## Exemplo - continuação

Pergunta: Quantos amigos Harry Potter tem?

BD: 2

## Exemplo - continuação

**Pergunta:** Quantos amigos Harry Potter tem?

BD: 2

Ontologia: pelo menos 1 (não UNA).

## Exemplo - continuação

Pergunta: Quantos amigos Harry Potter tem?

BD: 2

Ontologia: pelo menos 1 (não UNA).

Se acrescentarmos um fato novo:

$\neg(\text{ronWeasley}=\text{hermioneGranger})$

Pergunta: Quantos amigos Harry Potter tem?

## Exemplo - continuação

**Pergunta:** Quantos amigos Harry Potter tem?

BD: 2

Ontologia: pelo menos 1 (não UNA).

Se acrescentarmos um fato novo:

$\neg(\text{ronWeasley}=\text{hermioneGranger})$

**Pergunta:** Quantos amigos Harry Potter tem?

BD: 2



## Exemplo - continuação

**Pergunta:** Quantos amigos Harry Potter tem?

BD: 2

Ontologia: pelo menos 1 (não UNA).

Se acrescentarmos um fato novo:

$\neg(\text{ronWeasley}=\text{hermioneGranger})$

**Pergunta:** Quantos amigos Harry Potter tem?

BD: 2

Ontologia: pelo menos 2 (mundo aberto).

## Exemplo - continuação

Novos fatos inseridos:

Phoenix(fawks)

hasPet(dumbledore,fawks)

## Exemplo - continuação

Novos fatos inseridos:

Phoenix(fawks)

hasPet(dumbledore,fawks)

DB: Rejeita inserção (domínio de hasPet é Human e Dumbledore não é Human - mundo fechado)

Ontologia: Infere que Dumbledore é humano. E também que é mago (pois tem uma Fênix)