Sistemas Baseados em Conhecimento
Aula 14

Renata Wassermann

renata@ime.usp.br

2019

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

$$
\begin{array}{rcl}
C, D & \longrightarrow & \mathsf{A} \mid \\
 & & \top \mid \\
 & & \bot \mid \\
 & & \neg C \mid \\
 & & C \sqcap D \mid \\
 & & C \sqcup D \mid \\
 & & \forall R.C \mid \\
 & & \exists R.C
\end{array}
$$

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

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

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 Σ se satisfaz $\mathcal{T}$ e $\mathcal{A}$.

Uma base de conhecimento Σ é *satisfatível* se admite um modelo.

Consequência Lógica

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

Consequência Lógica

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

$\exists$teaches.Course $\sqsubseteq$ GraduateStudent $\sqcup$ Professor

teaches(john, cs101)
Course(cs101)
$\neg$Professor(john)

Consequência Lógica

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

$\exists$teaches.Course $\sqsubseteq$ GraduateStudent $\sqcup$ Professor

teaches(john, cs101)
Course(cs101)
$\neg$Professor(john)

$\Sigma \models$GraduateStudent(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

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

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

Tipos de raciocínio

- Satisfatibilidade:
  Verificar se Σ tem um modelo
- Satisfatibilidade de Conceito (em relação a Σ):
  Checar se existe um modelo $\mathcal{I}$ de Σ tal que $C^{\mathcal{I}} \neq \emptyset$
  $\Sigma \not\models C \equiv \bot$
  Student $\sqcap$ $\neg$Person

Tipos de raciocínio

- "Subsunção" (em relação a Σ):
  Checar se para todos os modelos $\mathcal{I}$ de Σ, $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$.
  $\Sigma \models C \sqsubseteq D$
  Student $\sqsubseteq$Person

Foi o primeiro problema estudado (classificação).
Para lógicas simples, resolvido por métodos estruturais.

Tipos de raciocínio

- Checagem de instância (em relação a Σ):
  Checar se para todo modelo $\mathcal{I}$ de Σ, $a^{\mathcal{I}} \in C^{\mathcal{I}}$.
  $\Sigma \models C(a)$

Tipos de raciocínio

- Checagem de instância (em relação a Σ):
  Checar se para todo modelo $\mathcal{I}$ de Σ, $a^{\mathcal{I}} \in C^{\mathcal{I}}$.
  $\Sigma \models C(a)$
- Recuperação
  Encontrar os indivíduos a tais que para todos os modelos $\mathcal{I}$ de Σ, $a^{\mathcal{I}} \in C^{\mathcal{I}}$
  $\{a | \Sigma \models C(a)\}$

Tipos de raciocínio

- Checagem de instância (em relação a Σ):
  Checar se para todo modelo $\mathcal{I}$ de Σ, $a^{\mathcal{I}} \in C^{\mathcal{I}}$.
  $\Sigma \models C(a)$
- Recuperação
  Encontrar os indivíduos a tais que para todos os modelos $\mathcal{I}$ de Σ, $a^{\mathcal{I}} \in C^{\mathcal{I}}$
  $\{a | \Sigma \models C(a)\}$
- Realização
  Encontrar os conceitos C tais que para todos os modelos $\mathcal{I}$ de Σ, $a^{\mathcal{I}} \in C^{\mathcal{I}}$
  $\{C | \Sigma \models C(a)\}$

Tipos de raciocínio

Todos os tipos podem ser vistos como satisfatibilidade.

Tipos de raciocínio

Todos os tipos podem ser vistos como satisfatibilidade.

Satisfatibilidade de conceito: $\Sigma \not\models C \equiv \bot$

Tipos de raciocínio

Todos os tipos podem ser vistos como satisfatibilidade.

Satisfatibilidade de conceito: $\Sigma \not\models C \equiv \bot$

sse

$\Sigma \cup \{C(x)\}$ é satisfazível.

Tipos de raciocínio

Todos os tipos podem ser vistos como satisfatibilidade.

Satisfatibilidade de conceito: $\Sigma \not\models C \equiv \bot$

sse

$\Sigma \cup \{C(x)\}$ é satisfazível.

Subsunção: $\Sigma \models C \sqsubseteq D$

Tipos de raciocínio

Todos os tipos podem ser vistos como satisfatibilidade.

Satisfatibilidade de conceito: $\Sigma \not\models C \equiv \bot$

sse

$\Sigma \cup \{C(x)\}$ é satisfazível.

Subsunção: $\Sigma \models C \sqsubseteq D$

sse

$\Sigma \cup \{(C \sqcap \neg D)(x)\}$ não é satisfazível.

Exemplo

Para saber se

$$\exists R.A \sqcap \exists R.B \sqsubseteq \exists R.(A \sqcap B)$$

devemos verificar se o conceito

$$\exists R.A \sqcap \exists R.B \sqcap \neg\exists R.(A \sqcap B)$$

é satisfazível, ou seja, se existe uma interpretação $\mathcal{I}$ tal que

$$(\exists R.A \sqcap \exists R.B \sqcap \neg\exists R.(A \sqcap B))^{\mathcal{I}} \neq \emptyset$$

Exemplo – solução

- Primeiro passo: forma normal negada
  $C \equiv \exists R.A \sqcap \exists R.B \sqcap \forall R.(\neg A \sqcup \neg B)$

Exemplo – solução

- Primeiro passo: forma normal negada
  $C \equiv \exists R.A \sqcap \exists R.B \sqcap \forall R.(\neg A \sqcup \neg B)$
- Se existir $\mathcal{I}$ tal que $C^{\mathcal{I}} \neq \emptyset$, então o axioma inicial não é válido.

Exemplo – solução

- Primeiro passo: forma normal negada
  $C \equiv \exists R.A \sqcap \exists R.B \sqcap \forall R.(\neg A \sqcup \neg B)$
- Se existir $\mathcal{I}$ tal que $C^{\mathcal{I}} \neq \emptyset$, então o axioma inicial não é válido.
- Inventa b e diz que $b \in C^{\mathcal{I}}$

Exemplo – solução

- Primeiro passo: forma normal negada
  $C \equiv \exists R.A \sqcap \exists R.B \sqcap \forall R.(\neg A \sqcup \neg B)$
- Se existir $\mathcal{I}$ tal que $C^{\mathcal{I}} \neq \emptyset$, então o axioma inicial não é válido.
- Inventa b e diz que $b \in C^{\mathcal{I}}$
- $b \in (\exists R.A)^{\mathcal{I}}$, logo, existe c tal que $(b, c) \in R^{\mathcal{I}}$ e $c \in A^{\mathcal{I}}$

Exemplo – solução

- Primeiro passo: forma normal negada
  $C \equiv \exists R.A \sqcap \exists R.B \sqcap \forall R.(\neg A \sqcup \neg B)$
- Se existir $\mathcal{I}$ tal que $C^{\mathcal{I}} \neq \emptyset$, então o axioma inicial não é válido.
- Inventa b e diz que $b \in C^{\mathcal{I}}$
- $b \in (\exists R.A)^{\mathcal{I}}$, logo, existe c tal que $(b, c) \in R^{\mathcal{I}}$ e $c \in A^{\mathcal{I}}$
- $b \in (\exists R.B)^{\mathcal{I}}$, logo, existe d tal que $(b, d) \in R^{\mathcal{I}}$ e $d \in B^{\mathcal{I}}$

Exemplo – solução

- Primeiro passo: forma normal negada
  $C \equiv \exists R.A \sqcap \exists R.B \sqcap \forall R.(\neg A \sqcup \neg B)$
- Se existir $\mathcal{I}$ tal que $C^{\mathcal{I}} \neq \emptyset$, então o axioma inicial não é válido.
- Inventa b e diz que $b \in C^{\mathcal{I}}$
- $b \in (\exists R.A)^{\mathcal{I}}$, logo, existe c tal que $(b, c) \in R^{\mathcal{I}}$ e $c \in A^{\mathcal{I}}$
- $b \in (\exists R.B)^{\mathcal{I}}$, logo, existe d tal que $(b, d) \in R^{\mathcal{I}}$ e $d \in B^{\mathcal{I}}$
- $b \in (\forall R.(\neg A \sqcup \neg B))^{\mathcal{I}}$, logo, $c, d \in (\neg A \sqcup \neg B)^{\mathcal{I}}$

Exemplo – solução

- Primeiro passo: forma normal negada
  $C \equiv \exists R.A \sqcap \exists R.B \sqcap \forall R.(\neg A \sqcup \neg B)$
- Se existir $\mathcal{I}$ tal que $C^{\mathcal{I}} \neq \emptyset$, então o axioma inicial não é válido.
- Inventa b e diz que $b \in C^{\mathcal{I}}$
- $b \in (\exists R.A)^{\mathcal{I}}$, logo, existe c tal que $(b, c) \in R^{\mathcal{I}}$ e $c \in A^{\mathcal{I}}$
- $b \in (\exists R.B)^{\mathcal{I}}$, logo, existe d tal que $(b, d) \in R^{\mathcal{I}}$ e $d \in B^{\mathcal{I}}$
- $b \in (\forall R.(\neg A \sqcup \neg B))^{\mathcal{I}}$, logo, $c, d \in (\neg A \sqcup \neg B)^{\mathcal{I}}$
- $c \in (\neg A)^{\mathcal{I}}$ gera conflito, logo $c \in (\neg B)^{\mathcal{I}}$

Exemplo – solução

- Primeiro passo: forma normal negada
  $C \equiv \exists R.A \sqcap \exists R.B \sqcap \forall R.(\neg A \sqcup \neg B)$
- Se existir $\mathcal{I}$ tal que $C^{\mathcal{I}} \neq \emptyset$, então o axioma inicial não é válido.
- Inventa b e diz que $b \in C^{\mathcal{I}}$
- $b \in (\exists R.A)^{\mathcal{I}}$, logo, existe c tal que $(b, c) \in R^{\mathcal{I}}$ e $c \in A^{\mathcal{I}}$
- $b \in (\exists R.B)^{\mathcal{I}}$, logo, existe d tal que $(b, d) \in R^{\mathcal{I}}$ e $d \in B^{\mathcal{I}}$
- $b \in (\forall R.(\neg A \sqcup \neg B))^{\mathcal{I}}$, logo, $c, d \in (\neg A \sqcup \neg B)^{\mathcal{I}}$
- $c \in (\neg A)^{\mathcal{I}}$ gera conflito, logo $c \in (\neg B)^{\mathcal{I}}$
- $d \in (\neg B)^{\mathcal{I}}$ gera conflito, logo $d \in (\neg A)^{\mathcal{I}}$

Exemplo – solução

Construímos uma interpretação $\mathcal{I}$:

- $\Delta^{\mathcal{I}} = \{b, c, d\}$

Exemplo – solução

Construímos uma interpretação $\mathcal{I}$:

- $\Delta^{\mathcal{I}} = \{b, c, d\}$
- $R^{\mathcal{I}} = \{(b, c), (b, d)\}$

Exemplo – solução

Construímos uma interpretação $\mathcal{I}$:

- $\Delta^{\mathcal{I}} = \{b, c, d\}$
- $R^{\mathcal{I}} = \{(b, c), (b, d)\}$
- $A^{\mathcal{I}} = \{c\}$

Exemplo – solução

Construímos uma interpretação $\mathcal{I}$:

- $\Delta^{\mathcal{I}} = \{b, c, d\}$
- $R^{\mathcal{I}} = \{(b, c), (b, d)\}$
- $A^{\mathcal{I}} = \{c\}$
- $B^{\mathcal{I}} = \{d\}$

Exemplo – solução

Construímos uma interpretação $\mathcal{I}$:

- $\Delta^{\mathcal{I}} = \{b, c, d\}$
- $R^{\mathcal{I}} = \{(b, c), (b, d)\}$
- $A^{\mathcal{I}} = \{c\}$
- $B^{\mathcal{I}} = \{d\}$

Exemplo – solução

Construímos uma interpretação $\mathcal{I}$:

- $\Delta^{\mathcal{I}} = \{b, c, d\}$
- $R^{\mathcal{I}} = \{(b, c), (b, d)\}$
- $A^{\mathcal{I}} = \{c\}$
- $B^{\mathcal{I}} = \{d\}$

$$(\exists R.A \sqcap \exists R.B \sqcap \forall R.(\neg A \sqcup \neg B))^{\mathcal{I}} \neq \emptyset$$

Exercício

Verificar se o seguinte axioma é válido:

$$\exists R.(A \sqcap B) \sqsubseteq \exists R.A \sqcap \exists R.B$$

$\mathcal{ALC}$

$$
\begin{array}{lcl}
C, D & \longrightarrow & \mathsf{A} \mid \\
 & & \top \mid \\
 & & \bot \mid \\
 & & \neg C \mid \\
 & & C \sqcap D \mid \\
 & & C \sqcup D \mid \\
 & & \forall R.C \mid \\
 & & \exists R.C
\end{array}
$$

$\mathcal{EL}$

$$\begin{array}{rcl} C, D & \longrightarrow & \mathsf{A} \mid \\ & & \top \mid \\ & & C \sqcap D \mid \\ & & \exists R.C \end{array}$$

Parece bastante inexpressiva, mas: SNOMED, Galen, etc.

$\mathcal{DL}$-Lite

$$
\begin{array}{lcl}
B & \longrightarrow & \mathrm{A} \mid \\
 & & \top \mid \\
 & & \exists R \mid \\
 & & \exists R^{-1} \text{ papéis inversos}
\end{array}
$$

$\mathcal{DL}$-Lite

$$
\begin{array}{lll}
B & \longrightarrow & \mathsf{A} \mid \\
 & & \top \mid \\
 & & \exists R \mid \\
 & & \exists R^{-1} \text{ papéis inversos (isPetOf} = \text{hasPet}^{-1}\text{)}
\end{array}
$$

$\mathcal{DL}$-Lite

$$
\begin{array}{lll}
B & \longrightarrow & \mathsf{A} \mid \\
 & & \top \mid \\
 & & \exists R \mid \\
 & & \exists R^{-1} \text{ papéis inversos (isPetOf} = \text{hasPet}^{-1}\text{)}
\end{array}
$$

$$
\begin{array}{lll}
C, D & \longrightarrow & \mathsf{B} \mid \\
 & & \neg B \mid \\
 & & C \sqcap D
\end{array}
$$

$\mathcal{SHIF}$

S $= \mathcal{ALC}+$ papéis transitivos
H $=$ hierarquia de papéis
I $=$ papéis inversos
F $=$ papéis funcionais

$\mathcal{SHIF}$

S $= \mathcal{ALC}+$ papéis transitivos (ancestor)
H $=$ hierarquia de papéis
I $=$ papéis inversos
F $=$ papéis funcionais

$\mathcal{SHIF}$

S = $\mathcal{ALC}+$ papéis transitivos (ancestor)
H = hierarquia de papéis (hasSon$\sqsubseteq$hasChild)
I = papéis inversos
F = papéis funcionais

$\mathcal{SHIF}$

S = $\mathcal{ALC}+$ papéis transitivos (ancestor)
H = hierarquia de papéis (hasSon$\sqsubseteq$hasChild)
I = papéis inversos (isPetOf = hasPet^{-1})
F = papéis funcionais

$\mathcal{SHIF}$

S = $\mathcal{ALC}+$ papéis transitivos (ancestor)
H = hierarquia de papéis (hasSon$\sqsubseteq$hasChild)
I = papéis inversos (isPetOf = hasPet^{-1})
F = papéis funcionais (isFatherOf)

$\mathcal{SHOIN}$

S $= \mathcal{ALC}+$ papéis transitivos
H $=$ hierarquia de papéis
O $=$ nominais
I $=$ papéis inversos
N $=$ restrições numéricas

$\mathcal{SHOIN}$

S = $\mathcal{ALC}+$ papéis transitivos
H = hierarquia de papéis
O = nominais ({hogwarts})
I = papéis inversos
N = restrições numéricas

$\mathcal{SHOIN}$

S = $\mathcal{ALC}+$ papéis transitivos
H = hierarquia de papéis
O = nominais ($\{$hogwarts$\}$)
I = papéis inversos
N = restrições numéricas ($>$ 3hasChild)

$\mathcal{SROIQ}$

S $=$ $\mathcal{ALC}+$ papéis transitivos
R $=$ cadeia de papéis $+$ hierarquias

O $=$ nominais
I $=$ papéis inversos
Q $=$ restrições numéricas qualificadas

$\mathcal{SROIQ}$

S = $\mathcal{ALC}$+ papéis transitivos
R = cadeia de papéis + hierarquias (hasParent $\circ$ hasBrother $\sqsubseteq$ hasUncle)
O = nominais
I = papéis inversos
Q = restrições numéricas qualificadas

$\mathcal{SROIQ}$

S = $\mathcal{ALC}+$ papéis transitivos
R = cadeia de papéis + hierarquias (hasParent $\circ$ hasBrother $\sqsubseteq$ hasUncle)
O = nominais
I = papéis inversos
Q = restrições numéricas qualificadas ($>$ 3hasChild.Male)