

Ontologías y Web Semántica

Clase 2-Parte3: Description Logic

Laura Cecchi Germán Braun

`{lcecchi,german.braun}@fi.uncoma.edu.ar`

Depart. de Teoría de la Computación - Facultad de Informática
Universidad Nacional del Comahue

Marzo 2020

1 Representación del Conocimiento

2 DL Básica: \mathcal{ALC}

- Sintaxis
- Semantics

3 Bibliografía

The role of Knowledge Representation

- Provee una caracterización precisa de una base de conocimiento, lo que involucra caracterizar el **tipo de conocimiento** a ser especificado por el sistema, como también definir claramente los **servicios de razonamiento** que el sistema necesita proveer (clases de consultas que el sistema debería responder).
- Énfasis en la **SEMÁNTICA** de los datos: fundamental para compartir, entender y razonar sobre los datos.

Una lógica permite la axiomatización de la información del dominio y la derivación de conclusiones desde esa información (conocimiento implícito).

- Sintaxis
- Semántica
- Inferencia Lógica = Razonamiento

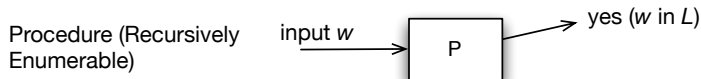
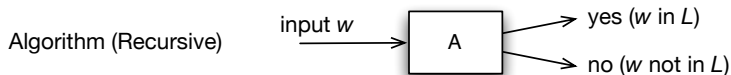
Representación del Conocimiento

Al elegir un lenguaje de representación de conocimiento hay algunas preguntas que debemos hacernos:

- Expressive Power of representation language: able to represent the problem
- Correctness of entailment procedure: no false conclusions are drawn
- Completeness of entailment procedure: all correct conclusions are drawn
- Decidability of entailment problem: there exists a (terminating) algorithm to compute entailment
- Complexity: resources needed for computing the solution

¿Por qué Description Logic?

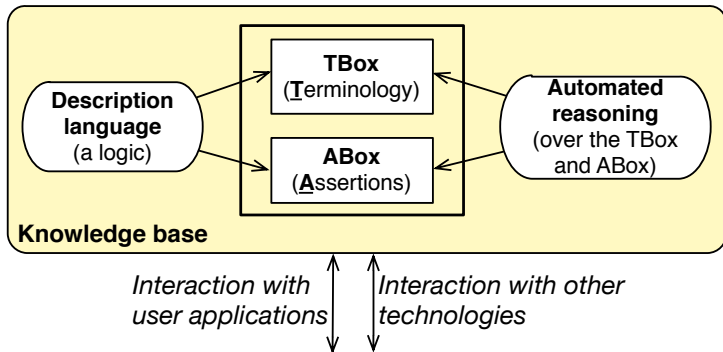
- Si acabamos de ver FOL, ¿por qué usar otra lógica?
- FOL es indecidible, lo que es una MALA NOTICIA



¿Qué es DLs?

- Es una familia de lógicas.
- Un fragmento estructurado de FOL
- Notación diferente pero mismas ideas de FOL
- Description Logic es un subconjunto de \mathcal{L}_3 , i.e., FOL libre de funciones utilizando únicamente como máximo tres nombres de variables.
- La representación es a nivel de predicados: ninguna variable está presente en la notación
- Provee teoría para **expresar información estructurada** en forma declarativa y para **acceder** y **razonar** sobre ella.

Description Logic knowledge base



La especificación de una base de conocimiento comprende dos niveles diferentes:

- **Nivel Intensional**: especifica el conjunto de elementos conceptuales y las restricciones/axiomas que describen la estructura conceptual del dominio del problema.

TBox(taxonomía) contruido a través de declaraciones que describen propiedades generales de los conceptos. Introduce la terminología,i.e. el vocabulario del dominio de aplicación.

- **Nivel Extensional**: especifica un conjunto de instancias de los elementos conceptuales descritos en el nivel intensional
ABox contiene las aserciones sobre individuos nombrados en término del vocabulario. Depende de un conjunto de circunstancias, variando en el tiempo.

Sintaxis de \mathcal{ALC} : Alfabeto

- **Conceptos** denotan tipos de entidades/clases/predicados unarios/universales, incluyendo top \top y bottom \perp ;
Ej.: (primitivas, atómicas): `Book`, `Course`
- **Roles** denotan relaciones/asociaciones/predicados n-arios/propiedades;
Ej.¹: `ENROLLED`, `READS`
- **Constructores**: ‘and’ \sqcap , ‘or’ \sqcup , y ‘not’ \neg ; cuantificadores ‘for all’ (cada) \forall y ‘exists’ (al menos uno/algún) \exists
- **Individuos** (objetos)
Ej.: `Student(Mandla)`, `Mother(Sally)`, \neg `Student(Sally)`, `ENROLLED(Mandla, CS101/19/2)`

¹Las mayúsculas en los roles es por claridad, pero no se requiere

Conceptos Complejos utiliza los constructores

Sean C y D conceptos, R un nombre de rol, luego

- $\neg C$, $C \sqcap D$, y $C \sqcup D$ son conceptos, y
- $\forall R.C$ y $\exists R.C$ son conceptos

Ej.:

- $\exists \text{ENROLLED.Course}$
- $\text{Woman} \sqcap \exists \text{PARENTOF.Person}$

Resumiendo: Lenguaje Básico AL

Construct	Syntax	Example
atomic concept	A	Doctor
atomic role	P	hasChild
atomic negation	$\neg A$	\neg Doctor
conjunction	$C \sqcap D$	Hum \sqcap Male
(unqual.) exist. res.	$\exists R$	\exists hasChild
value restriction	$\forall R.C$	\forall hasChild.Male
bottom	\perp	

Siendo C, D conceptos y R un rol.

Imagen extraída del Curso de Alessandro Artale.

Otros Constructores

- Disjunction: $\forall \text{hasChild} . (\text{Doctor} \sqcup \text{Lawyer})$
- Qualified existential restriction: $\exists \text{hasChild} . \text{Doctor}$
- Full negation: $\neg(\text{Doctor} \sqcup \text{Lawyer})$
- Number restrictions: $(\geq 2 \text{ hasChild}) \sqcap (\leq 1 \text{ sibling})$
- Qualified number restrictions: $(\geq 2 \text{ hasChild} . \text{Doctor})$
- Inverse role: $\forall \text{hasChild}^{-} . \text{Doctor}$
- Reflexive-transitive role closure: $\exists \text{hasChild}^{*} . \text{Doctor}$

Familia de Description Logics

Construct	\mathcal{AL}	Syntax
disjunction	\mathcal{U}	$C \sqcup D$
top		\top
qual. exist. res.	\mathcal{E}	$\exists R.C$
(full) negation	\mathcal{C}	$\neg C$
number	\mathcal{N}	$(\geq k R)$
restrictions		$(\leq k R)$
qual. number	\mathcal{Q}	$(\geq k R.C)$
restrictions		$(\leq k R.C)$
inverse role	\mathcal{I}	R^-
role closure	reg	R^*

Siendo C, D conceptos y R un rol.

Definición

Tienen la forma

$$C \sqsubseteq D \quad (R \sqsubseteq S)$$

llamados **inclusión** o

$$C \equiv D \quad (R \equiv S)$$

llamados **equivalencias**.

Siendo C y D conceptos y R y S roles.

Axiomas Terminológicos

Definición

Tienen la forma $C \sqsubseteq D$ ($R \sqsubseteq S$) llamados **inclusión** o $C \equiv D$ ($R \equiv S$) llamados **equivalencias**.

Ejemplos

- Todo administrador es un empleado

$$\textit{Administrador} \sqsubseteq \textit{Empleado}$$

- Una mujer es una persona de sexo femenino

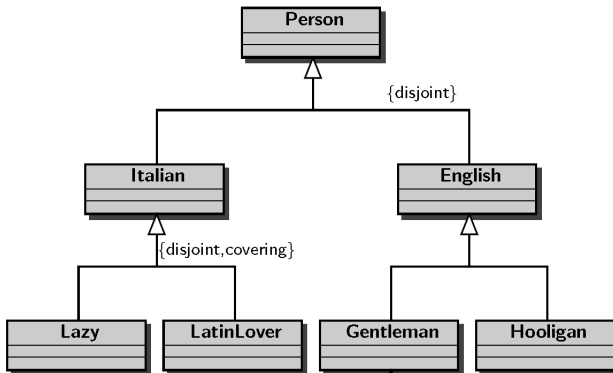
$$\textit{Woman} \equiv \textit{Person} \sqcap \textit{Female}$$

- Un hombre es una persona que no es una mujer.

$$\textit{Man} \equiv \textit{Person} \sqcap \neg \textit{Woman}$$

Ejercicio

Codifique el siguiente UML en DL.



Basada en la teoría de modelos.

Interpretación en \mathcal{ALC}

Una interpretación $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ consiste de

- Un conjunto no vacío de objetos $\Delta^{\mathcal{I}}$ llamado *dominio de la interpretación*
- Una *función de la interpretación* $\cdot^{\mathcal{I}}$ que mapea:
 - Cada concepto atómico A a un subconjunto $A^{\mathcal{I}}$ de $\Delta^{\mathcal{I}}$, i.e. $A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$
 - Cada role R a un subconjunto $R^{\mathcal{I}}$ de $\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$, i.e. $R^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$
 - Cada nombre de individuo a a un elemento de $\Delta^{\mathcal{I}}$, i. e. $a^{\mathcal{I}} \in \Delta^{\mathcal{I}}$
- Note: $\top^{\mathcal{I}} = \Delta^{\mathcal{I}}$ and $\perp^{\mathcal{I}} = \emptyset$

- $(\neg C)^{\mathcal{I}} = \Delta^{\mathcal{I}} \setminus C^{\mathcal{I}}$
- $(C \sqcap D)^{\mathcal{I}} = C^{\mathcal{I}} \cap D^{\mathcal{I}}$
- $(C \sqcup D)^{\mathcal{I}} = C^{\mathcal{I}} \cup D^{\mathcal{I}}$
- $(\forall R.C)^{\mathcal{I}} = \{o \mid \forall o'. (o, o') \in R^{\mathcal{I}} \rightarrow o' \in C^{\mathcal{I}}\}$
- $(\exists R.)^{\mathcal{I}} = \{o \mid \exists o'. (o, o') \in R^{\mathcal{I}}\}$

Sean C y D conceptos, R un rol, y a y b individuos

- Una interpretación \mathcal{I} satisface la sentencia $C \sqsubseteq D$ si

$$C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$$

- Una interpretación \mathcal{I} satisface la sentencia $C \equiv D$ si $C^{\mathcal{I}} = D^{\mathcal{I}}$
- $C(a)$ es satisfecho por \mathcal{I} si $a^{\mathcal{I}} \in C^{\mathcal{I}}$
- $R(a, b)$ es satisfecho por \mathcal{I} si $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in R^{\mathcal{I}}$

Resumiendo:

Construct	Syntax	Example	Semantics
atomic concept	A	Doctor	$A^I \subseteq \Delta^I$
atomic role	P	hasChild	$P^I \subseteq \Delta^I \times \Delta^I$
atomic negation	$\neg A$	\neg Doctor	$\Delta^I \setminus A^I$
conjunction	$C \sqcap D$	Hum \sqcap Male	$C^I \cap D^I$
(unqual.) exist. res.	$\exists R$	\exists hasChild	$\{ o \mid \exists o'. (o, o') \in R^I \}$
value restriction	$\forall R.C$	\forall hasChild.Male	$\{ o \mid \forall o'. (o, o') \in R^I \rightarrow o' \in C^I \}$
bottom	\perp		\emptyset

Imagen extraída del Curso de Alessandro Artale.

Resumiendo:

Construct	\mathcal{AL}	Syntax	Semantics
disjunction	\mathcal{U}	$C \sqcup D$	$C^{\mathcal{I}} \cup D^{\mathcal{I}}$
top		\top	$\Delta^{\mathcal{I}}$
qual. exist. res.	\mathcal{E}	$\exists R.C$	$\{ o \mid \exists o'. (o, o') \in R^{\mathcal{I}} \wedge o' \in C^{\mathcal{I}} \}$
(full) negation	\mathcal{C}	$\neg C$	$\Delta^{\mathcal{I}} \setminus C^{\mathcal{I}}$
number restrictions	\mathcal{N}	$(\geq k R)$	$\{ o \mid \#\{o' \mid (o, o') \in R^{\mathcal{I}}\} \geq k \}$
		$(\leq k R)$	$\{ o \mid \#\{o' \mid (o, o') \in R^{\mathcal{I}}\} \leq k \}$
qual. number restrictions	\mathcal{Q}	$(\geq k R.C)$	$\{ o \mid \#\{o' \mid (o, o') \in R^{\mathcal{I}} \wedge o' \in C^{\mathcal{I}}\} \geq k \}$
		$(\leq k R.C)$	$\{ o \mid \#\{o' \mid (o, o') \in R^{\mathcal{I}} \wedge o' \in C^{\mathcal{I}}\} \leq k \}$
inverse role	\mathcal{I}	R^{-}	$\{ (o, o') \mid (o', o) \in R^{\mathcal{I}} \}$
role closure	reg	R^{*}	$(R^{\mathcal{I}})^{*}$

Imagen extraída del Curso de Alessandro Artale.

Modelo de un Concepto

Una interpretación $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ es un **modelo** de un concepto C si $C^{\mathcal{I}} \neq \emptyset$.

Modelo de una Base de Conocimiento

Una interpretación $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ es un **modelo** de la base de conocimiento \mathcal{KB} si cada axioma de \mathcal{KB} es satisfecho por \mathcal{I} .

Satisfacible

Una base de conocimiento \mathcal{KB} se dice **satisfacible** si admite un modelo.

<http://www.cs.man.ac.uk/~ezolin/dl/>

Ejercicios

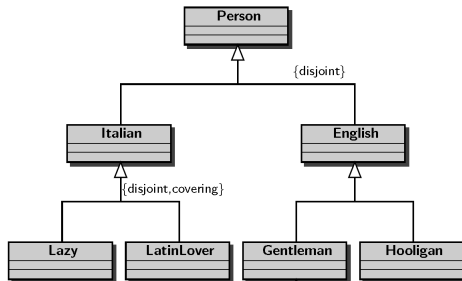
- 1 ¿Cuántos elementos tiene el dominio más chico que pueda tener una interpretación?
- 2 ¿Puede un dominio de interpretación ser infinito?
- 3 Demuestre que $A \sqcap \neg A$ no tiene modelos

Ejercicio

Suponga un dominio $\Delta^{\mathcal{I}} =$

$\{Williams, Andrea Bocelli, Simone Perrotta, Elton John, David Beckham, Pierluigi Casiraghi, DiegoMaradona\}$

Encuentre una interpretación para su representación en DL.



¿Es un modelo para Hooligan? ¿y para LatinLover?

¿Es un modelo para la base de conocimiento?

Invente un individuo de modo tal que pueda generar una interpretación que no sea modelo.



An Introduction to Ontology Engineering. v1

Keet, C. Maria - 2020



Formal Ontology, Conceptual Analysis and Knowledge Representation - Nicola Guarino

International journal of human-computer studies- 43(5) -Elsevier - 1995



Representing and Reasoning over a Taxonomy of Part-Whole Relations. Keet, C.M., Artale, A. *Applied Ontology*, 2008, 3(1-2):91-110.



The Description Logic Handbook: Theory, Implementation and Applications.

Franz Baader, Diego Calvanese, Deborah McGuinness, Daniele Nardi, and Peter F. Patel-Schneider, editors.

Cambridge University Press, 2003.



What is an ontology? Guarino N, Oberle D, Staab S. In Handbook on ontologies 2009 (pp. 1-17). Springer, Berlin, Heidelberg.

¿Preguntas?

¡Descanso! ¿?