

Lógicas de Descripción

Pedro Cabalar

Lógica

Grado en Inteligencia Artificial
Universidade da Coruña

April 8, 2025

Lógica de Descripción

- Para expresar conocimiento de un dominio usamos muchos **conceptos** relacionados entre sí.

Lógica de Descripción

- Para expresar conocimiento de un dominio usamos muchos **conceptos** relacionados entre sí.
- P.ej. un **pistón** **es una** pieza mecánica **y es parte del** motor que **es un** mecanismo ...

Lógica de Descripción

- Para expresar conocimiento de un dominio usamos muchos **conceptos** relacionados entre sí.
- P.ej. un **pistón** **es una** pieza mecánica y **es parte del** motor que **es un** mecanismo ...
- 👍 Representar **definiciones de conceptos** para un dominio dado = **Conocimiento Terminológico**
- Antiguamente se denominaban **sistemas terminológicos** o **lenguajes de conceptos**

Lógica de Descripción

- Para expresar conocimiento de un dominio usamos muchos **conceptos** relacionados entre sí.
- P.ej. un **pistón** **es una** pieza mecánica y **es parte del** motor que **es un** mecanismo ...
- 👍 Representar **definiciones de conceptos** para un dominio dado = **Conocimiento Terminológico**
- Antiguamente se denominaban **sistemas terminológicos** o **lenguajes de conceptos**
- Hoy en día: descripción de **Ontologías** para la **Web Semántica** (OWL-DL, OWL-Lite).

Lógica de Descripción

- Para expresar conocimiento de un dominio usamos muchos **conceptos** relacionados entre sí.
- P.ej. un **pistón** **es una** pieza mecánica y **es parte del** motor que **es un** mecanismo ...
- 👍 Representar **definiciones de conceptos** para un dominio dado = **Conocimiento Terminológico**
- Antiguamente se denominaban **sistemas terminológicos** o **lenguajes de conceptos**
- Hoy en día: descripción de **Ontologías** para la **Web Semántica** (OWL-DL, OWL-Lite).
- **Lógica de Descripción** (*Description Logic*, DL) usa operadores básicos (similar a lógica modal) que se pueden traducir a **cálculo de predicados**.

- **Sintaxis:** se forma a partir de
 - ① Un conjunto de átomos llamados **conceptos**. Representan **clases** o **conjuntos** de objetos. Ej.: *Persona*, *Mamífero*, *Vehículo*, *Azul*, ...

Lógica de Descripción

- **Sintaxis:** se forma a partir de
 - ① Un conjunto de átomos llamados **conceptos**. Representan **clases** o **conjuntos** de objetos. Ej.: *Persona*, *Mamífero*, *Vehículo*, *Azul*, ...
 - ② Un conjunto de **roles** que representan **relaciones binarias** entre objetos. Ej.: *le_gusta*, *posee*, *viaja_en*, *cría* ...

- **Sintaxis:** se forma a partir de
 - 1 Un conjunto de átomos llamados **conceptos**. Representan **clases** o **conjuntos** de objetos. Ej.: *Persona*, *Mamífero*, *Vehículo*, *Azul*, ...
 - 2 Un conjunto de **roles** que representan **relaciones binarias** entre objetos. Ej.: *le_gusta*, *posee*, *viaja_en*, *cría* ...
 - 3 Un conjunto de **constructores** que definen nuevos conceptos recursivamente. Ej.: vehículos rojos o azules = $Vehículo \sqcap (Azul \sqcup Rojo)$.
Ej.: un *Padre* es un $Hombre \sqcap \exists cría$

Lógica de Descripción

- **Sintaxis**: se forma a partir de
 - 1 Un conjunto de átomos llamados **conceptos**. Representan **clases** o **conjuntos** de objetos. Ej.: *Persona*, *Mamífero*, *Vehículo*, *Azul*, ...
 - 2 Un conjunto de **roles** que representan **relaciones binarias** entre objetos. Ej.: *le_gusta*, *posee*, *viaja_en*, *cría* ...
 - 3 Un conjunto de **constructores** que definen nuevos conceptos recursivamente. Ej.: vehículos rojos o azules = $Vehículo \sqcap (Azul \sqcup Rojo)$.
Ej.: un *Padre* es un $Hombre \sqcap \exists cría$
- Existe toda una **clasificación** o familia de Lógicas de Descripción distintas dependiendo de los constructores que permitimos usar

- La DL más sencilla es \mathcal{FL}^- . Sintaxis:

$$C ::= A \mid C \sqcap D \mid \forall R.C \mid \exists R \mid \top \mid \perp$$

donde A =concepto atómico, C, D =conceptos y R =rol.

- La DL más sencilla es \mathcal{FL}^- . Sintaxis:

$$C ::= A \mid C \sqcap D \mid \forall R.C \mid \exists R \mid \top \mid \perp$$

donde A =concepto atómico, C, D =conceptos y R =rol.

- Sintaxis alternativa:

$$C ::= A \mid (: \text{and } C D) \mid (: \text{all } R C) \mid (: \text{some } R)$$

- La DL más sencilla es \mathcal{FL}^- . Sintaxis:

$$C ::= A \mid C \sqcap D \mid \forall R.C \mid \exists R \mid \top \mid \perp$$

donde A =concepto atómico, C, D =conceptos y R =rol.

- Sintaxis alternativa:

$$C ::= A \mid (: \text{and } C D) \mid (: \text{all } R C) \mid (: \text{some } R)$$

- Cuantificadores: $\forall \text{cría.Hembra}$ son los seres vivos cuyas crías son todas hembras, mientras que $\exists \text{cría}$ son aquellos que tienen al menos una cría.

- La DL más sencilla es \mathcal{FL}^- . Sintaxis:

$$C ::= A \mid C \sqcap D \mid \forall R.C \mid \exists R \mid \top \mid \perp$$

donde A =concepto atómico, C, D =conceptos y R =rol.

- Sintaxis alternativa:

$$C ::= A \mid (: \text{and } C D) \mid (: \text{all } R C) \mid (: \text{some } R)$$

- Cuantificadores: $\forall \text{cría.Hembra}$ son los seres vivos cuyas crías son todas hembras, mientras que $\exists \text{cría}$ son aquellos que tienen al menos una cría. Podemos escribir esto último como $\exists \text{cría}.\top$ too

Definición (Interpretación)

Una *interpretación* \mathcal{I} es un par $\langle \Delta^{\mathcal{I}}, \cdot^{\mathcal{I}} \rangle$ donde

- $\Delta^{\mathcal{I}}$ es un conjunto no vacío llamado *dominio*
- $\cdot^{\mathcal{I}}$ es una función que hace corresponder:
 - 1 Cada concepto con un subconjunto de $\Delta^{\mathcal{I}}$.
 - 2 Cada rol con un subconjunto de $\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$.

Definición (Interpretación)

Una *interpretación* \mathcal{I} es un par $\langle \Delta^{\mathcal{I}}, \cdot^{\mathcal{I}} \rangle$ donde

- $\Delta^{\mathcal{I}}$ es un conjunto no vacío llamado *dominio*
- $\cdot^{\mathcal{I}}$ es una función que hace corresponder:
 - 1 Cada concepto con un subconjunto de $\Delta^{\mathcal{I}}$.
 - 2 Cada rol con un subconjunto de $\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$.

- Ampliamos la función $\cdot^{\mathcal{I}}$ para evaluar conceptos no atómicos del siguiente modo:

$$(C \sqcap D)^{\mathcal{I}} = C^{\mathcal{I}} \cap D^{\mathcal{I}}$$

$$(\exists R)^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} : \exists y.(x, y) \in R^{\mathcal{I}}\}$$

$$(\forall R.C)^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} : \forall y.(x, y) \in R^{\mathcal{I}} \Rightarrow y \in C^{\mathcal{I}}\}$$

$$(\top)^{\mathcal{I}} = \Delta^{\mathcal{I}}$$

$$(\perp)^{\mathcal{I}} = \emptyset$$

- Ampliamos la función $\cdot^{\mathcal{I}}$ para evaluar conceptos no atómicos del siguiente modo:

$$(C \sqcap D)^{\mathcal{I}} = C^{\mathcal{I}} \cap D^{\mathcal{I}}$$

$$(\exists R)^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} : \exists y.(x, y) \in R^{\mathcal{I}}\}$$

$$(\forall R.C)^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} : \forall y.(x, y) \in R^{\mathcal{I}} \Rightarrow y \in C^{\mathcal{I}}\}$$

$$(\top)^{\mathcal{I}} = \Delta^{\mathcal{I}}$$

$$(\perp)^{\mathcal{I}} = \emptyset$$

- Ejemplos: ¿qué significan las siguientes expresiones?

Adulto \sqcap *Macho*

- Ampliamos la función $\cdot^{\mathcal{I}}$ para evaluar conceptos no atómicos del siguiente modo:

$$(C \sqcap D)^{\mathcal{I}} = C^{\mathcal{I}} \cap D^{\mathcal{I}}$$

$$(\exists R)^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} : \exists y.(x, y) \in R^{\mathcal{I}}\}$$

$$(\forall R.C)^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} : \forall y.(x, y) \in R^{\mathcal{I}} \Rightarrow y \in C^{\mathcal{I}}\}$$

$$(\top)^{\mathcal{I}} = \Delta^{\mathcal{I}}$$

$$(\perp)^{\mathcal{I}} = \emptyset$$

- Ejemplos: ¿qué significan las siguientes expresiones?

Adulto \sqcap *Macho*

$\forall \text{cría}.(Adulto \sqcap Macho)$

- Ampliamos la función $\cdot^{\mathcal{I}}$ para evaluar conceptos no atómicos del siguiente modo:

$$(C \sqcap D)^{\mathcal{I}} = C^{\mathcal{I}} \cap D^{\mathcal{I}}$$

$$(\exists R)^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} : \exists y.(x, y) \in R^{\mathcal{I}}\}$$

$$(\forall R.C)^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} : \forall y.(x, y) \in R^{\mathcal{I}} \Rightarrow y \in C^{\mathcal{I}}\}$$

$$(\top)^{\mathcal{I}} = \Delta^{\mathcal{I}}$$

$$(\perp)^{\mathcal{I}} = \emptyset$$

- Ejemplos: ¿qué significan las siguientes expresiones?

Adulto \sqcap *Macho*

$\forall \text{cría}.$ (*Adulto* \sqcap *Macho*)

$\exists \text{cría} \sqcap \forall \text{cría}.$ ($\exists \text{cría} \sqcap$ *Adulto*)

- Un segundo ejemplo: \mathcal{ALC} es más expresiva que \mathcal{FL}^- . Permite:

$$C ::= A \mid C \sqcap D \mid \forall R.C \mid \exists R \mid \neg C$$

$$\text{con } (\neg C)^{\mathcal{I}} = \Delta^{\mathcal{I}} \setminus C^{\mathcal{I}}$$

- Un segundo ejemplo: \mathcal{ALC} es más expresiva que \mathcal{FL}^- . Permite:

$$C ::= A \mid C \sqcap D \mid \forall R.C \mid \exists R \mid \neg C$$

con $(\neg C)^{\mathcal{I}} = \Delta^{\mathcal{I}} \setminus C^{\mathcal{I}}$

- Una variante más limitada, \mathcal{AL} , reemplaza $\neg C$ por $\neg A$ (sólo se pueden negar **conceptos** atómicos).

Lógica de Descripciones: constructores más habituales

Constructor	Sintaxis	Semántica
Aserción de Concepto	$c : C$	$c^I \in C^I$
Aserción de Rol	$(o_1, o_2) : R$	$(o_1^I, o_2^I) \in R^I$
Intersección	$C \sqcap D$	$C^I \cap D^I$
Unión	$C \sqcup D$	$C^I \cup D^I$
Negación	$\neg C$	$\Delta^I \setminus C^I$
Existencial	$\exists R.C$	$\{x : \exists y.(x, y) \in R^I \& y \in C^I\}$
Universal	$\forall R.C$	$\{x : \forall y.(x, y) \in R^I \Rightarrow y \in C^I\}$
Cardinalidad	$(\geq n R)$	$\{x : \{y.(x, y) \in R^I\} \geq n\}$
	$(\leq n R)$	$\{x : \{y.(x, y) \in R^I\} \leq n\}$
Inverso	R^-	$\{(x, y) : (y, x) \in R^I\}$
Transitivo	R^*	$(R^I)^*$
Enumeración (uno-de)	$\{o_1, \dots, o_n\}$	$\{o_1^I, \dots, o_n^I\}$

...

Lógica de Descripciones: nomenclatura

- Para nombrar cada combinación se suelen usar las **iniciales**
 - \mathcal{F} propiedad funcional ($\leq 1 R$)
 - \mathcal{E} existencial general $\exists R.C$
 - \mathcal{U} unión $C \sqcup D$
 - \mathcal{C} negación $\neg C$
 - $\mathcal{S} = \mathcal{ALC}$ con roles transitivos R^*
 - \mathcal{H} jerarquía de roles
 - \mathcal{R} roles reflexivos, irreflexivos u disjuntos
 - \mathcal{I} rol inverso R^-
 - \mathcal{O} enumeración (uno-de) $\{o_1, \dots, o_n\}$
 - \mathcal{N} restricciones de cardinalidad ($\geq n R$), ($\leq n R$)
 - \mathcal{Q} restricciones de cardinalidad cualificadas ($\geq n R.C$), ($\leq n R.C$)
 - (\mathcal{D}) tipos de datos (enteros, strings, etc)

Lógica de Descripciones: nomenclatura

- Para nombrar cada combinación se suelen usar las **iniciales**
 - \mathcal{F} propiedad funcional ($\leq 1 R$)
 - \mathcal{E} existencial general $\exists R.C$
 - \mathcal{U} unión $C \sqcup D$
 - \mathcal{C} negación $\neg C$
 - $\mathcal{S} = \mathcal{ALC}$ con roles transitivos R^*
 - \mathcal{H} jerarquía de roles
 - \mathcal{R} roles reflexivos, irreflexivos u disjuntos
 - \mathcal{I} rol inverso R^-
 - \mathcal{O} enumeración (uno-de) $\{o_1, \dots, o_n\}$
 - \mathcal{N} restricciones de cardinalidad ($\geq n R$), ($\leq n R$)
 - \mathcal{Q} restricciones de cardinalidad cualificadas ($\geq n R.C$), ($\leq n R.C$)
 - (\mathcal{D}) tipos de datos (enteros, strings, etc)
- **OWL-DL** viene a ser $\mathcal{SHOIN}^{(\mathcal{D})}$ mientras que **OWL-Lite** está basado en $\mathcal{SHIF}^{(\mathcal{D})}$.

Lógica de Descripción

- Una **Base de Conocimiento** (terminológico) se compone de dos conjuntos de expresiones **TBox** y **ABox**.

Lógica de Descripción

- Una **Base de Conocimiento** (terminológico) se compone de dos conjuntos de expresiones **TBox** y **ABox**.
- **TBox**: contiene las declaraciones terminológicas generales. Dos tipos:
 - 1 Una **definición de concepto** $A \equiv C$. Ejemplos:

Mujer \equiv *Persona* \sqcap *Hembra*

Madre \equiv *Mujer* \sqcap \exists *cría*

(se suele requerir **definiciones no cíclicas**)

Lógica de Descripción

- Una **Base de Conocimiento** (terminológico) se compone de dos conjuntos de expresiones **TBox** y **ABox**.
- **TBox**: contiene las declaraciones terminológicas generales. Dos tipos:
 - 1 Una **definición de concepto** $A \equiv C$. Ejemplos:

$$\begin{aligned} \textit{Mujer} &\equiv \textit{Persona} \sqcap \textit{Hembra} \\ \textit{Madre} &\equiv \textit{Mujer} \sqcap \exists \textit{cría} \end{aligned}$$

(se suele requerir **definiciones no cíclicas**)

- 2 Un **axioma de inclusión** $C_1 \sqsubseteq C_2$. Ejemplo

$$\exists \textit{cría}.\textit{Persona} \sqsubseteq \textit{Persona}$$

Imponen restricciones en nuestro modelo

- **ABox**: contiene aserciones sobre elementos particulares del dominio o de relaciones entre ellos. Dos tipos:
 - 1 Aserción de concepto $o : C$. Ejemplo:

Moby_Dick : Ballena

María : Hembra $\sqcap \exists$ cría

- **ABox**: contiene aserciones sobre elementos particulares del dominio o de relaciones entre ellos. Dos tipos:

- 1 **Aserción de concepto** $o : C$. Ejemplo:

Moby_Dick : Ballena

María : Hembra \sqcap \exists cría

- 2 **Aserciones de Rol** $(o_1, o_2) : R$. Ejemplo:

(María, Jesús) : cría

- **Problemas de razonamiento** típicos para resolver. Dada una base de conocimiento:
 - 1 *Subsunción*. Comprobar si un concepto es más general que otro
 $C \sqsubseteq D$

Lógica de Descripción: razonamiento

- **Problemas de razonamiento** típicos para resolver. Dada una base de conocimiento:
 - 1 **Subsunción**. Comprobar si un concepto es más general que otro
 $C \sqsubseteq D$
 - 2 **Equivalencia**. Comprobar si dos conceptos son equivalentes $C \equiv D$

Lógica de Descripción: razonamiento

- **Problemas de razonamiento** típicos para resolver. Dada una base de conocimiento:
 - 1 **Subsunción**. Comprobar si un concepto es más general que otro
 $C \sqsubseteq D$
 - 2 **Equivalencia**. Comprobar si dos conceptos son equivalentes $C \equiv D$
 - 3 **Consistencia**. Comprobar si un concepto tiene al menos algún modelo $C \equiv \perp$

Lógica de Descripción: razonamiento

- **Problemas de razonamiento** típicos para resolver. Dada una base de conocimiento:
 - 1 **Subsunción**. Comprobar si un concepto es más general que otro $C \sqsubseteq D$
 - 2 **Equivalencia**. Comprobar si dos conceptos son equivalentes $C \equiv D$
 - 3 **Consistencia**. Comprobar si un concepto tiene al menos algún modelo $C \equiv \perp$
 - 4 **Pertenencia**. Comprobar si un individuo es miembro de un concepto $o : C$

Lógica de Descripción: razonamiento

- **Problemas de razonamiento** típicos para resolver. Dada una base de conocimiento:
 - 1 **Subsunción**. Comprobar si un concepto es más general que otro $C \sqsubseteq D$
 - 2 **Equivalencia**. Comprobar si dos conceptos son equivalentes $C \equiv D$
 - 3 **Consistencia**. Comprobar si un concepto tiene al menos algún modelo $C \equiv \perp$
 - 4 **Pertenencia**. Comprobar si un individuo es miembro de un concepto $o : C$
- Todos estos problemas se pueden **reducir** a consistencia.
Ejemplo: $C \sqsubseteq D$ se puede reducir a comprobar la consistencia de $C \sqcap \neg D \equiv \perp$.

Lógica de Descripción: reasoning

- Existen métodos eficientes para resolver estos problemas de razonamiento
- Normalmente, variante de lenguaje **más expresiva** \Rightarrow razonamiento asociado **más complejo**

- Existen métodos eficientes para resolver estos problemas de razonamiento
- Normalmente, variante de lenguaje **más expresiva** \Rightarrow razonamiento asociado **más complejo**
- Ejemplo: subsunción \mathcal{FL}^- es decidible en complejidad temporal P .
- Ver **Description Logic Complexity Navigator**
<http://www.cs.man.ac.uk/~ezolin/dl/>

Lógica de Descripción: traducción a Lógica de Primer Orden

- La mayoría de las Lógicas de Descripción son reducibles a fragmentos **decidibles** de Lógica de Primer Orden

Lógica de Descripción: traducción a Lógica de Primer Orden

- La mayoría de las Lógicas de Descripción son reducibles a fragmentos **decidibles** de Lógica de Primer Orden
- Cada concepto C se convierte en un predicado unario $C(x)$ y cada rol R en un predicado binario $R(x, y)$.

Lógica de Descripción: traducción a Lógica de Primer Orden

- La mayoría de las Lógicas de Descripción son reducibles a fragmentos **decidibles** de Lógica de Primer Orden
- Cada concepto C se convierte en un predicado unario $C(x)$ y cada rol R en un predicado binario $R(x, y)$.
- Podemos usar Lógica de Primer Orden con dos nombres de variable x, y :

$$\begin{array}{ll} t_x(A) = A(x) & t_y(A) = A(y) \\ t_x(C \sqcap D) = t_x(C) \wedge t_x(D) & t_y(C \sqcap D) = t_y(C) \wedge t_y(D) \\ t_x(C \sqcup D) = t_x(C) \vee t_x(D) & t_y(C \sqcup D) = t_y(C) \vee t_y(D) \\ t_x(\exists R.C) = \exists y.R(x, y) \wedge t_y(C) & t_y(\exists R.C) = \exists x.R(y, x) \wedge t_x(C) \\ t_x(\forall R.C) = \forall y.R(x, y) \Rightarrow t_y(C) & t_y(\forall R.C) = \forall x.R(y, x) \Rightarrow t_x(C) \end{array}$$

Lógica de Descripción: traducción a Lógica de Primer Orden

- La mayoría de las Lógicas de Descripción son reducibles a fragmentos **decidibles** de Lógica de Primer Orden
- Cada concepto C se convierte en un predicado unario $C(x)$ y cada rol R en un predicado binario $R(x, y)$.
- Podemos usar Lógica de Primer Orden con dos nombres de variable x, y :

$$\begin{array}{ll} t_x(A) = A(x) & t_y(A) = A(y) \\ t_x(C \sqcap D) = t_x(C) \wedge t_x(D) & t_y(C \sqcap D) = t_y(C) \wedge t_y(D) \\ t_x(C \sqcup D) = t_x(C) \vee t_x(D) & t_y(C \sqcup D) = t_y(C) \vee t_y(D) \\ t_x(\exists R.C) = \exists y.R(x, y) \wedge t_y(C) & t_y(\exists R.C) = \exists x.R(y, x) \wedge t_x(C) \\ t_x(\forall R.C) = \forall y.R(x, y) \Rightarrow t_y(C) & t_y(\forall R.C) = \forall x.R(y, x) \Rightarrow t_x(C) \end{array}$$

- En una TBox, traducimos $C \equiv D$ como $\forall x.t_x(C) \leftrightarrow t_x(D)$.