Web ngữ nghĩa

Soạn bởi: Nguyễn Bá Ngọc

Chương 6

Hà Nội-2021

Chương 6.
Ngữ nghĩa hình thức của OWL

Nội dung

6.1. Lô-gic bậc nhất
6.2. Lô-gic mô tả
6.3. Ngữ nghĩa lô-gic của OWL
6.4. Suy diễn tự động với OWL

Nội dung

6.1. Lô-gic bậc nhất

6.2. Lô-gic mô tả

6.3. Ngữ nghĩa lô-gic của OWL

6.4. Suy diễn tự động với OWL

Biểu diễn tri thức

Mục đích cơ bản:

Phát triển các nền tảng hình thức để tạo biểu diễn bậc cao về thế giới để có thể tính toán hiệu quả và sử dụng để xây dựng các ứng dụng thông minh

Các cách tiếp cận trực quan, ví dụ các cấu trúc mạng

Các cách tiếp cận dựa trên lô-gic, ví dụ phần khả tính của lô-gic bậc nhất (FOL) / Lô-gic mô tả

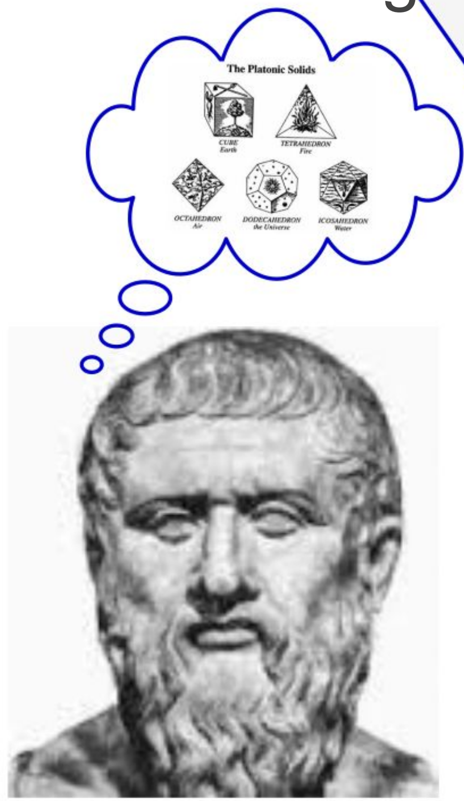
Lịch sử biểu diễn tri thức nhận biết

Plato: "***Knowledge*** *is justified true belief*"

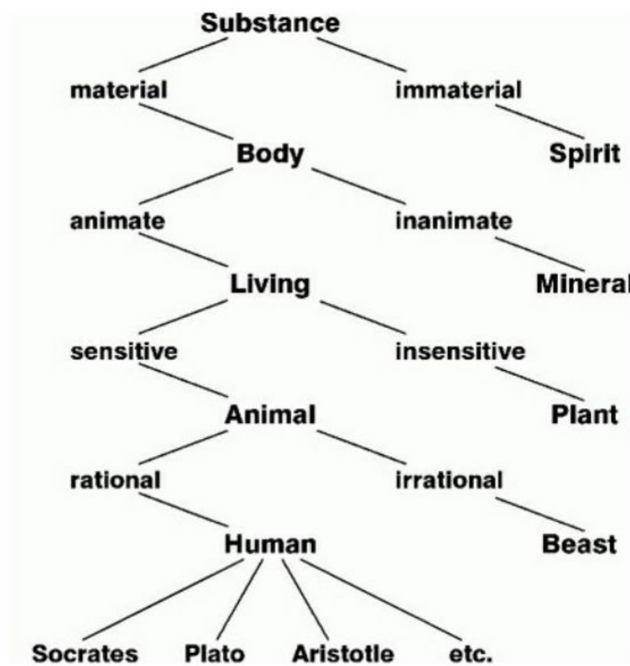
Tri thức là niềm tin đã được xác minh là đúng.

350 BC

Bên trong



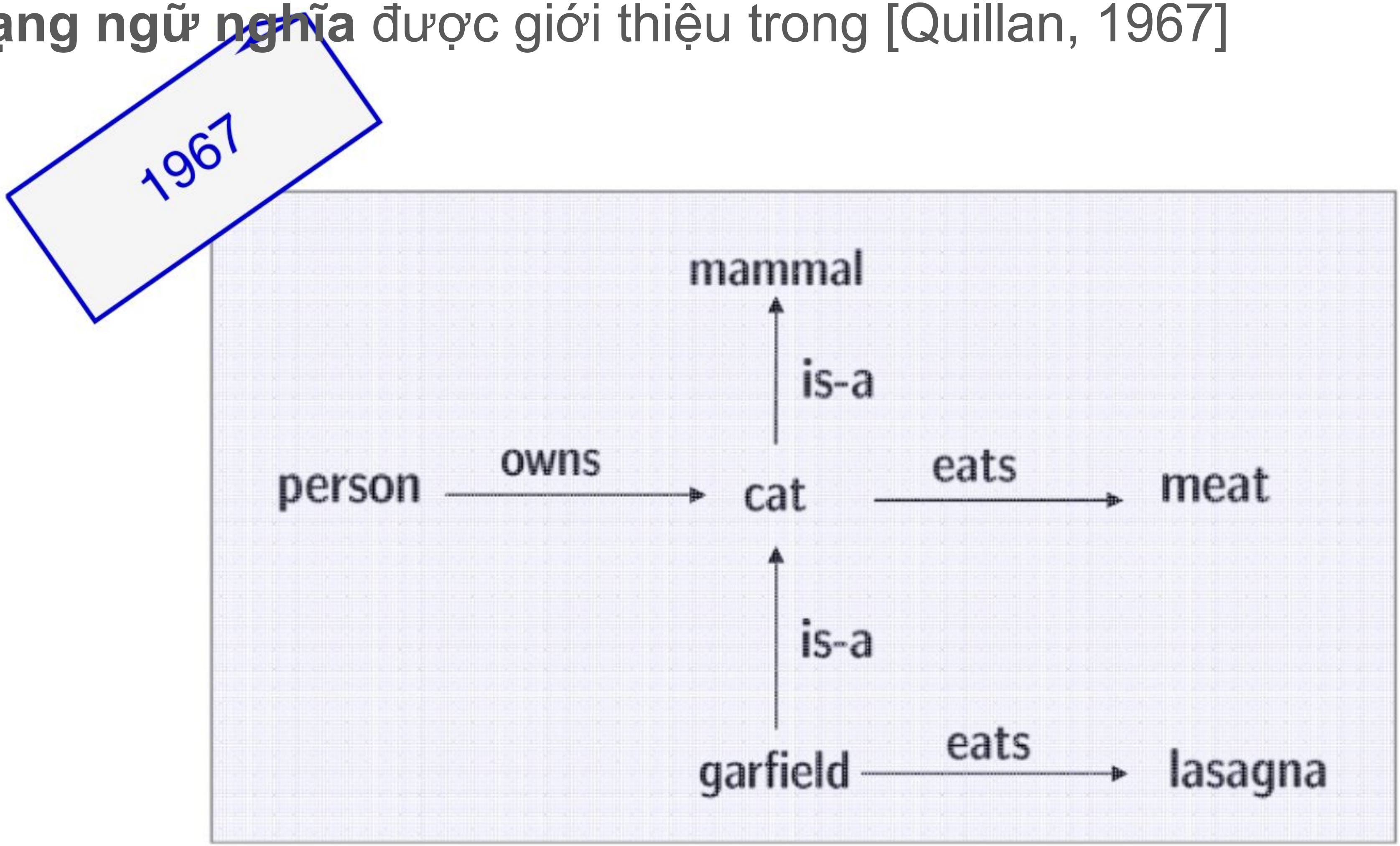
Bên ngoài



Cây tri thức

Lịch sử biểu diễn tri thức nhận biết$_{(2)}$

Mạng ngữ nghĩa được giới thiệu trong [Quillan, 1967]



Lịch sử biểu diễn tri thức dựa trên lô-gic

- Những năm 1950: Lô-gic bậc nhất cho biểu diễn tri thức (không khả quyết) [McCarthy, 1959]
- Những năm 1960: Các cấu trúc mạng cho biểu diễn tri thức (không có ngữ nghĩa hình thức) [Quillan, 1967]
- Những năm 1970: Biểu diễn cấu trúc mạng bằng FOL [Hayes, 1979].
- Những năm 1980:
  - Các khung lô-gic [Minsky, 1985]
  - Lô-gic mô tả (DL) cho biểu diễn tri thức
    - Các phần khả quyết của FOL;
    - Các tri thức được biểu diễn bằng DLs cũng được gọi là ontology;
    - Nhiều phiên bản DLs với khả năng diễn đạt và tính toán khác nhau;
    - Thường được sử dụng để suy diễn khái niệm;

Lô-gic mệnh đề và lô-gic bậc nhất

Có thể biểu diễn tri thức sau trong nền tảng hình thức nào?

(1) Aristotle là người   (2) Socrates là người

(3) Người không bất tử

=> Aristotle không bất tử và Socrates không bất tử.

- **Lô-gic vị từ/mệnh đề (PL - Propositional Logic)**: Các biến mệnh đề, ¬, ⋁, ⋀, →
  - AristotleIsAMan = true;
  - SocratesIsAMan = true
  - AristotleIsAMan →AristotleIsMortal
  - SocratesIsAMan→SocratesIsMortal

Khả năng diễn đạt của PL tương đối hạn chế ...

Lô-gic mệnh đề và lô-gic bậc nhất$_{(2)}$

Có thể biểu diễn tri thức sau trong nền tảng hình thức nào?

(1) Aristotle là người (2) Socrates là người

(3) Người không bất tử

=> Aristotle không bất tử và Socrates không bất tử.

- **Lô-gic bậc nhất (FOL - First Order Logic):** Các thuộc tính, hằng, biến, hàm ký tự, ¬, ∨, ∧, ∀, ∃, →
  - Man(Socrates);
  - Man(Aristotel);
  - ∀X(Man(X)) → Mortal(X)

FOL có khả năng diễn đạt cao nhưng không khả quyết trong trường hợp tổng quát.

Cú pháp của FOL

- FOL cung cấp những thành phần sau:
  - Các biến: x, y, …
  - Các toán tử: Tương tự như lô-gic vị từ/mệnh đề (PL): phủ định (¬), và (⊓), hoặc (⊔), suy ra (=>), khi và chỉ khi (<=>)
  - Định lượng: Tất cả (∀) và tồn tại (∃)
- Người dùng tự định nghĩa các thành phần sau:
  - Các ký hiệu: "Các phần tử" trong thế giới;
    - Ví dụ, Mary, 3, …
  - Các hàm: Ánh xạ các phần tử tới các phần tử;
    - Ví du, father-of(Mary) = John, color-of(Sky) = Blue
  - Các mệnh đề: Ánh xạ các thành phần tới các giá trị chân lý;
    - Ví dụ: greater(5, 3), green(Grass), color(Grass, Green)

Các định lượng

- Định lượng tất cả
  - Tương tự liên kết và ("and") cho tất cả thành phần
  - (∀x)P(x) - Với tất cả x mà P(x) đúng
    - Ví dụ, (∀x)dolphin(x) => mammal(x)
- Định lượng tồn tại
  - Tương tự liên kết hoặc ("or")
  - (∃x)P(x) - Tồn tại x mà P(x) đúng
    - Ví dụ, (∃x) mammal(x) ⊓ lays-eggs(x)
- Định lượng tất cả thường được sử dụng với liên kết "suy ra" để thiết lập luật nếu-thì
  - Ví dụ, (∀x) IT6390-student(x) => hard-working(x)
    - "Tất cả các học viên IT6390 đều chăm học"

Các định lượng$_{(2)}$

- Định lượng tồn tại thường được sử dụng với liên kết "và" để xác định các tính chất
    - Ví dụ, $(\exists x)$ IT6390-student(x) □ female(x)
        - "Có nữ học viên IT6390"
- Đảo trật tự các toán tử tất cả không làm thay đổi ý nghĩa
    - $(\forall x)(\forall y)P(x, y)$ tương đương với $(\forall y)(\forall x)P(x, y)$
    - Tương tự cũng có thể đảo trật tự các toán tử tồn tại
- Đảo trật tự kết hợp các định lượng tất cả và định lượng tồn tại cũng không làm thay đổi ý nghĩa
    - $(\forall x)(\exists y)$likes(x, y) tương đương với $(\exists y)(\forall x)$likes(x, y)
        - Tất cả mọi người đều thích ai đó

Cú pháp FOL

- Các câu được xây dựng dựa trên từ và các thành phần:
  - Từ: Chỉ 1 đối tượng trong thế giới thực, 1 hằng ký tự, 1 tên biến, hoặc 1 hàm;
    - Ví dụ, left-leg-of(); x và $f(x_1, ..., x_n)$ là các từ, trong đó x_i là từ
  - Thành phần: Có giá trị đúng hoặc sai
    - Nếu P và Q là các thành phần, thì ¬P, $P \sqcap Q$, $P \sqcup Q$, P=>Q, P <=> Q cũng là các thành phần.
- Câu có thể là:
  - Thành phần, hoặc
  - Với P là câu và x là biến, thì $(\forall x)P$ và $(\exists x)P$ cũng là câu
- Điều kiện diễn đạt đúng quy cách
  - Câu không chứa biến tự do
    - Các biến đều được gắn với các định lượng
    - Ví dụ $(\forall x)P(x, y)$ - x được gắn với định lượng tất cả, còn y là biến tự do.

Phiên dịch NNTN sang FOL

- Tất cả người làm vườn đều thích mặt trời
    - Every gardener likes the sun
    - (∀x) gardener(x) => likes (x, Sun)
- Bạn luôn có thể đánh lừa ai đó
    - You can fool some of the people all of the time
    - (∃x)(∀t) (person(x) ⊓ time(t)) => can-fool(x, t)
- Đôi khi bạn có thể nói dối tất cả mọi người
    - (∀x)(∃t) (person(x) ⊓ time(t)) => can-fool(x, t)
- Tất cả nấm mầu tím đều có độc
    - (∀x) (mushroom(x) ⊓ purple(x)) => poisonous (x)

Phiên dịch NNTN sang FOL (2)

- Không có nấm tím nào có độc
  - No purple mushroom is poisonous
  - ¬(∃x) purple(x) ⊓ mushroom(x) ⊓ poisonous(x), hoặc:
  - (∀x) (mushroom(x) ⊓ purple(x)) => ¬poisonous(x)
- Có đúng hai nấm màu tím
  - There are exactly two purple mushrooms
  - (∃x)(∃y) mushroom(x) ⊓ purple(x) ⊓ mushroom(y) ⊓ purple(y) ⊓ ¬(x=y) ⊓ (∀z) (mushroom(z) ⊓ purple(z)) => (x=z) ⊔ (y=z)
- NVA không cao:
  - NVA is not tall
  - ¬tall(NVA)

Phiên dịch NNTN sang FOL (3)

- X ở phía trên Y khi và chỉ khi X trực tiếp nằm trên Y hoặc có nhiều đối tượng được xếp chồng lên nhau bắt đầu với Y và kết thúc với X
  - $(\forall x)(\forall y)$ above (x, y) <=> (on(x, y) $\sqcup$ ($\exists$ z) (on(x, z) $\sqcap$ above(z, y)))

Suy diễn

- Suy diễn trong lô-gic hình thức là tiến trình sinh các câu hợp lệ từ những câu hợp lệ đang có (trong CSTT) bằng cách áp dụng các luật suy diễn
  - Luật suy diễn là đúng đắn nếu
    - Tất cả các câu X được sinh bởi luật suy diễn trên CSTT đều được suy ra theo lô-gic từ CSTT
    - Luật suy diễn không tạo ra mâu thuẫn
  - Luật suy diễn là đầy đủ nếu
    - Có thể sinh tất cả các câu có thể suy ra theo lô-gic từ CSTT

Vắn tắt về tính khả quyết

Tính khả quyết

Một lớp vấn đề được gọi là khả quyết nếu tồn tại giải thuật cho bất kỳ vấn đề nào thuộc lớp này như đầu vào, giải thuật có thể kiểm tra "đúng" hoặc "sai" trong thời gian hữu hạn.

Giải thuật có tính dừng

Lô-gic khả quyết

Trong phạm vi lô-gic vấn đề khái quát sau được nghiên cứu

Đầu vào: Một tập phát biểu $\mathcal{T}$ và phát biểu ϕ

Đầu ra: "Đúng" nếu $\mathcal{T}$ suy ra ϕ, và "Sai" nếu ngược lại.

Trong trường hợp không có nhập nhằng với vấn đề được xử lý, tùy theo khả năng giải quyết vấn đề lô-gic còn được gọi là khả quyết hoặc không khả quyết.

Vắn tắt về tính khả quyết (2)

Tính khả quyết của lô-gic mệnh đề

Xét lô-gic mệnh đề và các phát biểu $\mathcal{T}$ và ϕ như sau:

$$\underbrace{(\text{SocrIsAMan} \rightarrow \text{SocrIsMortal}) \land \text{SocrIsAMan}}_{\mathcal{T}} \underbrace{\vDash}_{\text{suy ra}} \underbrace{\text{SocrIsMortal}}_{\phi}$$

Các câu hỏi tương đương trong PL

- $\mathcal{T} \vDash \phi$? ;
- $\mathcal{T} \vDash \phi$ với tất cả các giá trị của *SocrIsAMan* và *SocrIsMortal*?;
- $\mathcal{T} \land \neg\phi$ không thỏa mãn - sai với tất cả các giá trị?

Vấn đề (không) được đáp ứng ((un)satisfiability) trong PL được gọi là (UN)SAT. Lô-gic mệnh đề là khả quyết, bởi vì (UN)SAT là khả quyết.

Nội dung

6.1. Lô-gic bậc nhất

6.2. Lô-gic mô tả

6.3. Ngữ nghĩa lô-gic của OWL

6.4. Suy diễn tự động với OWL

Lô-gic mô tả: Khái quát

- Phần khả quyết của lô-gic bậc nhất
- Một phần nghiên cứu DL đáng kể hướng tới làm rõ chi phí tính toán cho các bài toán suy diễn trong trường hợp độ phức tạp cao nhất.
- Tuy có độ phức tạp cao nhưng DLs cũng có khả năng diễn đạt cao, vẫn có các giải thuật suy diễn được tối ưu hóa có hành vi tốt trong môi trường ứng dụng
  - Ví dụ, SAT solving: Thuộc lớp NP-khó nhưng hoạt động tốt trong thực tế.

Lô-gic mô tả: Khái quát$_{(2)}$

- Lô-gic mô tả (DLs) là nền tảng biểu diễn tri thức tiêu biểu hiện nay
- Đã ảnh hưởng đáng kể tới sự hình thành của các ngôn ngữ Web ngữ nghĩa
  - OWL chủ yếu dựa trên lô-gic mô tả
- Có nhiều công cụ suy diễn

Jena Mastro HermiT FaCT++ Pellet ELK
...

Các thành phần DL

- Các phần tử: john, mary, sun
    - ⇝ các hằng trong FOL, các tài nguyên trong RDF
- Các khái niệm: Person, Course, Student, Film, v.v.
    - ⇝ các thuộc tính một đối số (FOL), các lớp (RDFS)
- Các vai trò: hasFather, attends, worksWith, v.v.
    - ⇝ các thuộc tính hai tham số (FOL), các thuộc tính (RDFS)
    - Có thể tiếp tục được chia nhỏ thành vai trò trừu tượng và cụ thể (các thuộc tính đối tượng và thuộc tính dữ liệu trong OWL).

Tập hợp tất cả các phần tử, khái niệm, và vai trò được gọi là chữ ký hoặc bộ từ vựng.

Các thành phần của CSTT DL

TBox $\mathcal{T}$	Biểu diễn các khái niệm và mối quan hệ giữa các khái niệm
ABox $\mathcal{A}$	Biểu diễn các phần tử, gắn kết các phần tử với các khái niệm, mối quan hệ giữa các phần tử

với khả năng diễn tả cao hơn DLs còn có:

RBox $\mathcal{R}$	Biểu diễn các vai trò và mối quan hệ giữa các vai trò

Các thành phần của DL

Một DL được xác định bởi:

- Ngôn ngữ mô tả: Làm cách nào để tạo các biểu thức khái niệm/vai trò, ví dụ:
  - Human $\sqcap$ Male $\sqcap$ $\exists$hasChild $\sqcap$ $\forall$hasChild.(Doctor $\sqcup$ Lawyer)
- Cơ chế mô tả các khái niệm (TBox $\mathcal{T}$) và các vai trò (RBox $\mathcal{R}$), ví dụ:
  - $\mathcal{T}$ = {Father $\equiv$ Human $\sqcap$ Male $\sqcap$ $\exists$hasChild, HappyFather $\sqsubseteq$ Father $\sqcap$ $\forall$hasChild.(Doctor $\sqcup$ Lawyer)}
  - $\mathcal{R}$ = {hasFather $\sqsubseteq$ hasParent}

Các thành phần của $DL_{(2)}$

- Cơ chế mô tả các thuộc tính của đối tượng (ABox $\mathcal{A}$)
  - $\mathcal{A}$ = {HappyFather (john), hasChild (john, mary)}
- Tập luật suy diễn: Suy diễn được gì trên CSTT?
  - $T \models \text{HappyFather} \sqcap \exists \text{hasChild}.(\text{Doctor} \sqcup \text{Lawyer})$
  - $T \cup A \models (\text{Doctor} \sqcup \text{Lawyer})(\text{mary})$

Các biểu thức khái niệm

$\mathcal{ALC}$ - ngôn ngữ thuộc tính với phần bù - Attributive Language with Complement, là hình thức DL cơ bản nhất thuộc lớp Boolean đóng.

Biểu thức khái niệm trong $\mathcal{ALC}$ được thiết lập như sau:

- Tên khái niệm đồng thời là biểu thức khái niệm
- $\top$ và $\bot$ là các biểu thức khái niệm
- Với $a_1, \ldots, a_n$ là các phần tử, $\{a_1, \ldots, a_n\}$ là một biểu thức khái niệm
- C và D là các khái niệm, $\neg C$, $C \sqcap D$, and $C \sqcup D$ là các biểu thức khái niệm

Các biểu thức khái niệm$_{(2)}$

- Với 1 vai trò r và 1 khái niệm C, $\exists r.C$ và $\forall r.C$ là các biểu thức khái niệm.
  - Ví dụ: $Student \sqcup \forall attendsCourse.MasterCourse$
- Với 1 vai trò đơn giản s, 1 biểu thức khái niệm C, và 1 số nguyên n, $\exists s.Self$ và $\leq n\ s.C$ và $\geq n\ s.C$ là các biểu thức khái niệm.

* Ghi chú: Chúng ta định nghĩa các vai trò và vai trò đơn giản sau (ở thời điểm hiện tại chúng ta sử dụng tên vai trò)

Ví dụ 6.1. Biểu thức khái niệm

- Và - Conjunction: $Singer \sqcap Actor$
- Hoặc - Disjunction: $\forall hasChild.(Doctor \sqcup Lawyer)$
- Giới hạn với định lượng tồn tại: $\exists hasChild.Doctor$
- Phủ định/Phần bù - Negation: $\neg(Doctor \sqcup Lawyer)$
- Giới hạn số lượng: $(\geq 2\ hasChild) \sqcap (\leq 1\ sibling)$
- Giới hạn số lượng định kiểu: $(\geq 2\ hasChild.Doctor)$
- Vai trò ngược: $\forall hasChild^{-}.Doctor$

TBox

Mệnh đề bao gồm khái niệm (*GCI - General Concept Inclusion*) có dạng:

$C \sqsubseteq D$, trong đó C và D là các khái niệm

- $C \equiv D$ là viết tắt của $C \sqsubseteq D$ và $D \sqsubseteq C$
- TBox $\mathcal{T}$ (Hộp thuật ngữ) bao gồm một tập GCIs

Lưu ý: Định nghĩa TBox giả định RBox đã được biết do ràng buộc vai trò đơn giản.

TBox $\mathcal{T}$

ABox

Mỗi khẳng định $\mathcal{ALC}$ có thể thuộc 1 trong các dạng sau:

- C(a), được gọi là khẳng định khái niệm
- r(a, b) - khẳng định vai trò
- ¬r(a, b) - phủ định vai trò
- a ≈ b - phát biểu tương đương
- a ≉ b - phát biểu không tương đương (khác nhau)

ABox bao gồm 1 tập các khẳng định

ABox $\mathcal{A}$

Mệnh đề bao gồm vai trò

- Một vai trò có thể là
  - Một tên vai trò r hoặc
  - Tên vai trò nghịch đảo r^- (đảo vị trí các thành phần), hoặc
  - vai trò toàn thể u
- Mỗi mệnh đề bao gồm vai trò (RIA) là 1 khẳng định dạng
  - $r_1 \circ ... \circ r_n \sqsubseteq r$, trong đó $r_1, ..., r_n, r$ là các vai trò
- $r \equiv s$ là viết tắt của $r \sqsubseteq s$ và $s \sqsubseteq r$

Tính đơn giản của vai trò

- Cho các RIAs, các vai trò được chia thành đơn giản và phức tạp
- Một cách tương đối, vai trò là phức tạp nếu có thể xuất hiện trong rhs (vế phải) của 1 RIA phức tạp
- Cụ thể hơn:
  - Với bất kỳ RIA $r_1 \circ r_2 \circ ... \circ r_n \sqsubseteq r$ với $n > 1$, r là phức tạp
  - Với bất kỳ RIA $s \sqsubseteq r$ với s phức tạp, thì r phức tạp và
  - Tất cả các thuộc tính còn lại là đơn giản

Ví dụ: $q \circ p \sqsubseteq r$ $\quad r \circ p \sqsubseteq r$ $\quad r \sqsubseteq s$ $\quad p \sqsubseteq r$ $\quad q \sqsubseteq s$

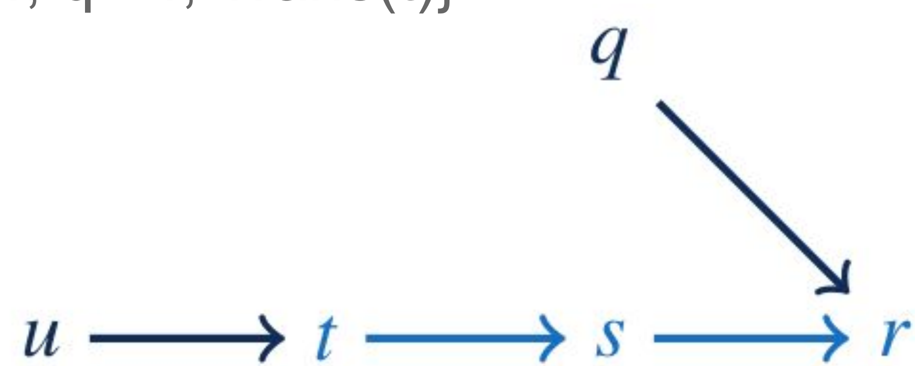
Phức tạp: r, s $\qquad$ Đơn giản: p, q

Tính đơn giản của vai trò$_{(2)}$

- Cho một cây vai trò $\mathcal{R}$, ký hiệu $\sqsubseteq^*_{\mathcal{R}}$ là bao đóng phản xạ và bắc cầu theo $\sqsubseteq$.
- Đối với một cây vai trò $\mathcal{R}$, chúng ta có thể chia các vai trò trong $\mathcal{R}$ thành đơn giản và phức tạp
- Một vai trò r là phức tạp theo $\mathcal{R}$, nếu có 1 vai trò t sao cho Trans(t) $\in \mathcal{R}$ và $\sqsubseteq^*_{\mathcal{R}}$ thỏa mãn.
- Tất cả các vai trò khác là đơn giản.
- Ví dụ: $\mathcal{R} = \{u \sqsubseteq t, t \sqsubseteq s, s \sqsubseteq r, q \sqsubseteq r, \text{Trans}(t)\}$

Phức tạp: t, s, r

Đơn giản: q, u



RBox

- Một biểu thức vai trò không giao nhau có dạng
  - Dis(s_1, s_2), trong đó s_1 và s_2 là các vai trò đơn giản
- Trong DL có khả năng diễn tả cao, $\mathcal{R}$ còn có thể chứa các mệnh đề, như Asy(r) (bất đối xứng), và Ref (r) (phản xạ)
  - Dis, Asy, Ref được gọi là các đặc điểm của các vai trò.

RBox bao gồm 1 tập các RIAs và 1 tập đặc điểm của các vai trò.

RBox $\mathcal{R}$

Ví dụ 6.2. CSTT DL

RBox $\mathcal{R}$

own $\sqsubseteq$ careFor

"Nếu ai đó sở hữu cái gì thì họ quan tâm về nó"

TBox $\mathcal{T}$

Healthy $\sqsubseteq \neg$Dead

Cat $\sqsubseteq$ Dead $\sqcup$ Alive

HappyCatOwner $\sqsubseteq \exists$ owns.Cat $\sqcap \forall$ caresFor.Healthy

"Sống khỏe mạnh là không chết"; " Mèo có thể đã chết hoặc còn sống"; "Chủ nhân hạnh phúc của mèo là người sở hữu mèo và tất cả mèo mà người đó quan tâm đều khỏe mạnh."

ABox $\mathcal{A}$

HappyCatOwner(NVA)

"NVA là người sở hữu mèo hạnh phúc."

Ngữ nghĩa hình thức của DL

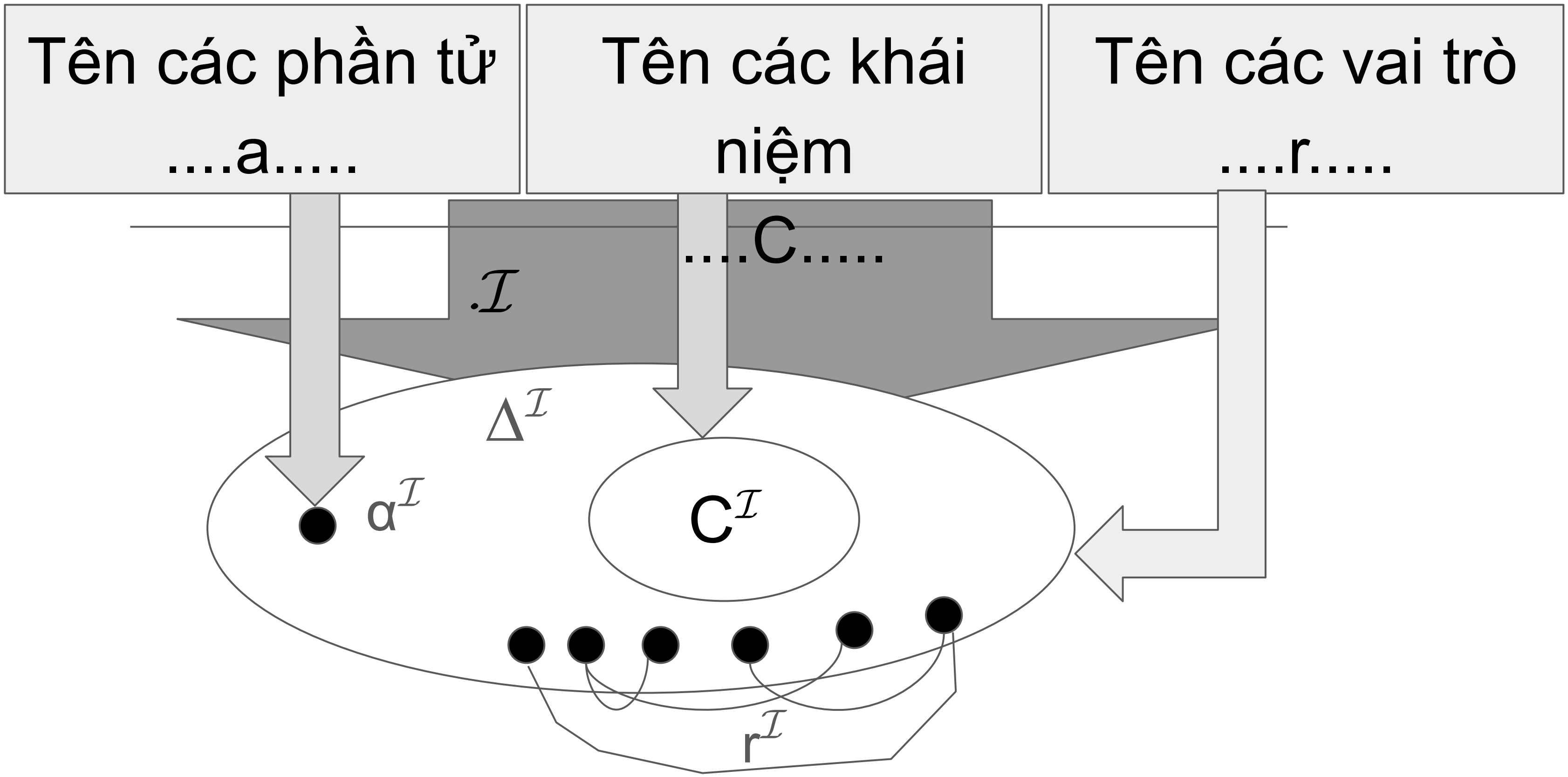
Các biểu diễn

- Ngữ nghĩa hình thức của DLs được xây dựng theo mô hình-lý thuyết dựa trên các không gian trừu tượng, được gọi là các biểu diễn.

Định nghĩa: Một biểu diễn $\mathcal{I} = (\Delta^{\mathcal{I}}, .^{\mathcal{I}})$ bao gồm:

- Một tập không rỗng $\Delta^{\mathcal{I}}$, được gọi là miền biểu diễn
- và một hàm $.^{\mathcal{I}}$ là ánh xạ
  - các tên riêng α tới các phần tử lĩnh vực $\alpha^{\mathcal{I}} \in \Delta^{\mathcal{I}}$
  - tên khái niệm C tới tập con $C^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$.
  - tên vai trò r tới các tập con $r^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$

Sơ đồ của 1 biểu diễn



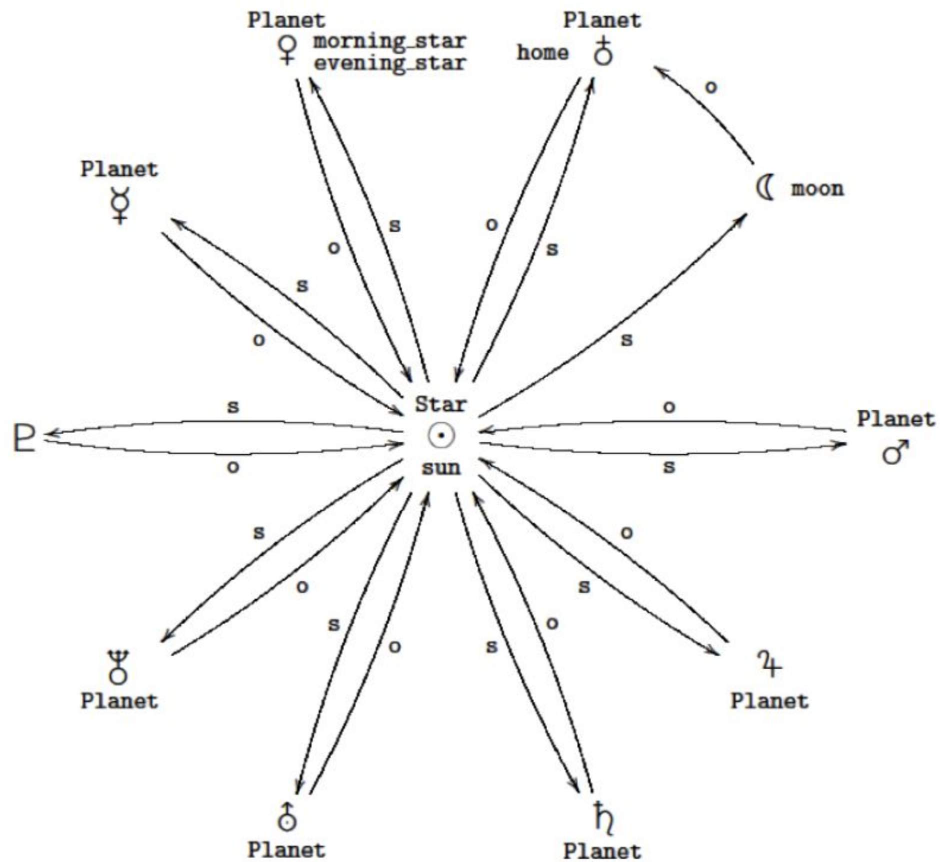
Ví dụ 6.3. Các biểu diễn

$\mathsf{N}_I = \{\mathtt{sun}, \mathtt{morning\_star}, \mathtt{evening\_star}, \mathtt{moon}, \mathtt{home}\}.$
$\mathsf{N}_C = \{\mathtt{Planet}, \mathtt{Star}\}.$
$\mathsf{N}_R = \{\mathtt{orbitsAround}, \mathtt{shinesOn}\}.$

$$\Delta^{\mathcal{I}} = \{☉, ☿, ♀, ♁, ☾, ♂, ♃, ♄, ⛢, ♅, ♇\}$$

$$\begin{aligned}
\mathtt{sun}^{\mathcal{I}} &= ☉ \\
\mathtt{morning_star}^{\mathcal{I}} &= ♀ \\
\mathtt{evening_star}^{\mathcal{I}} &= ♀ \\
\mathtt{moon}^{\mathcal{I}} &= ☾ \\
\mathtt{home}^{\mathcal{I}} &= ♁
\end{aligned}$$

$$\begin{aligned}
\mathtt{Planet}^{\mathcal{I}} &= \{☿, ♀, ♁, ♂, ♃, ♄, ⛢, ♅\} \\
\mathtt{Star}^{\mathcal{I}} &= \{☉\} \\
\mathtt{orbitsAround}^{\mathcal{I}} &= \{\langle ☿, ☉\rangle, \langle ♀, ☉\rangle, \langle ♁, ☉\rangle, \langle ♂, ☉\rangle, \langle ♃, ☉\rangle, \\
&\quad \langle ♄, ☉\rangle, \langle ⛢, ☉\rangle, \langle ♅, ☉\rangle, \langle ♇, ☉\rangle, \langle ☾, ♁\rangle\} \\
\mathtt{shinesOn}^{\mathcal{I}} &= \{\langle ☉, ☿\rangle, \langle ☉, ♀\rangle, \langle ☉, ♁\rangle, \langle ☉, ☾\rangle, \langle ☉, ♂\rangle, \\
&\quad \langle ☉, ♃\rangle, \langle ☉, ♄\rangle, \langle ☉, ⛢\rangle, \langle ☉, ♅\rangle, \langle ☉, ♇\rangle\}
\end{aligned}$$



Biểu diễn các phần tử

Giả thuyết tên duy nhất (UNA-Unique Name Assumption)

Với các định danh $c_1 \neq c_2$, thì $c^{\mathcal{I}}_1 \neq c^{\mathcal{I}}_2$

**Lưu ý:* Nếu UNA đúng thì các kiểm tra tương đương và khác biệt không còn cần thiết. Giả thuyết UNA có thể bị bác bỏ trong DLs.

Giả thuyết tên tiêu chuẩn (SNA-Standard Name Assumption)

Giả thuyết UNA đúng, và hơn nữa các phần tử được biểu diễn theo 1 cách trong tất cả các biểu diễn. Như vậy, chúng ta có thể giả sử $\Delta^{\mathcal{I}}$ chứa một tập các phần tử, và với mỗi biểu diễn $\mathcal{I}$, chúng ta có $c^{\mathcal{I}} = c$ (khi đó c được gọi là tên tiêu chuẩn).

Biểu diễn các biểu thức khái niệm

Tên	**Cú pháp**	**Ngữ nghĩa**
đỉnh - top	$\top$	$\Delta^{\mathcal{I}}$
đáy - bottom	$\bot$	$\varnothing$
phần bù - negation	$\neg C$	$\Delta^{\mathcal{I}} \setminus C^{\mathcal{I}}$
giao - conjunction	$C \sqcap D$	$C^{\mathcal{I}} \cap D^{\mathcal{I}}$
hợp - disjunction	$C \sqcup D$	$C^{\mathcal{I}} \cup D^{\mathcal{I}}$
định lượng tất cả - universal quantifier	$\forall r.C$	$\{x \in \Delta^{\mathcal{I}}$ | $(x, y) \in r^{\mathcal{I}}$ và $y \in C^{\mathcal{I}}\}$
định lượng tồn tại - existential quantifier	$\exists r.C$	$\{x \in \Delta^{\mathcal{I}}$ | có $y \in \Delta^{\mathcal{I}}$ thỏa mãn $(x, y) \in r^{\mathcal{I}}$ và $y \in C^{\mathcal{I}}\}$

Biểu diễn các biểu thức khái niệm$_{(2)}$

Cấu trúc	$\mathcal{AL}$	**Cú pháp**	**Ngữ nghĩa**	**Ghi chú**
Giới hạn số lượng	$\mathcal{N}$	$(\geq k\ r)$ $(\leq k\ r)$	≥ 2 hasSister ≤ 3 hasBrother	$\{o \mid \#\{o' \mid (o, o') \in r^{\mathcal{I}}\} \geq k\}$ $\{o \mid \#\{o' \mid (o, o') \in r^{\mathcal{I}}\} \leq k\}$
Giới hạn số lượng định kiểu	$\mathcal{Q}$	$(\geq k\ r.C)$ $(\leq k\ r.C)$	≥ 2 hasSibling.F ≤ 3 hasSibling.M	$\{o \mid \#\{o' \mid (o, o') \in r^{\mathcal{I}} \wedge o' \in C^{\mathcal{I}}\} \geq k$ $\{o \mid \#\{o' \mid (o, o') \in r^{\mathcal{I}} \wedge o' \in C^{\mathcal{I}}\} \leq k$

Nhiều cấu trúc DL và các tổ hợp của chúng đã được nghiên cứu.
Bằng cách kết hợp các cấu trúc khác nhau chúng ta thu được các phần DL khác nhau.

Biểu diễn các biểu thức vai trò

Cấu trúc	**Cú pháp**	**Ví dụ**	**Ngữ nghĩa**
Vai trò hạt nhân	r	hasChild	$r^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$
Phần bù vai trò	$\neg r$	¬hasSister	$\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}} \setminus \{(o, o') \in r^{\mathcal{I}}\}$
Vai trò ngược	r^{-}	hasParent⁻	$\{(o, o') \mid (o', o) \in r^{\mathcal{I}}\}$
Chuỗi vai trò	$r \circ r'$	hasChild ∘ hasParent	$\{(o, o') \mid (o, o'') \in r^{\mathcal{I}}, (o'', o') \in r'^{\mathcal{I}}\}$

Ngữ nghĩa của các mệnh đề

Kiểm tra biểu diễn $\mathcal{I}$ có phải là mô hình của mệnh đề α hay không (ký hiệu là: $\mathcal{I} \vDash \alpha$)

Tên	**Cú pháp**	**Ngữ nghĩa**	**Ký hiệu**
bao gồm	$C \sqsubseteq D$	đúng nếu $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$	$\mathcal{I} \vDash C \sqsubseteq D$
tương đương	$C \equiv D$	đúng nếu $C^{\mathcal{I}} = D^{\mathcal{I}}$	$\mathcal{I} \vDash C \equiv D$
kiểm tra khái niệm	$C(a)$	đúng nếu $a^{\mathcal{I}} \in C^{\mathcal{I}}$	$\mathcal{I} \vDash C(a)$
kiểm tra vai trò	$r(a, b)$	đúng nếu $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in r^{\mathcal{I}}$	$\mathcal{I} \vDash r(a, b)$
phủ định vai trò	$\neg r(a, b)$	đúng nếu $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in r^{\mathcal{I}}$	$\mathcal{I} \vDash r(a, b)$

Biến đổi DL về FOL

Biểu thức DL	Biểu thức FOL
$A \sqsubseteq B$	$\forall x\, A(x) \rightarrow B(x)$
$p \sqsubseteq q$	$\forall x, y\; p(x, y) \rightarrow q(x, y)$
$A \sqsubseteq \neg B$	$\forall x\, A(x) \rightarrow \neg B(x)$
$p \sqsubseteq \neg q$	$\forall x, y\; p(x, y) \rightarrow \neg q(x, y)$
$\exists r \sqsubseteq A$	$\forall x\; \exists y\; r(x, y) \rightarrow A(x)$
$\exists r^{-} \sqsubseteq A$	$\forall y\; \exists x\; r(x, y) \rightarrow A(y)$
$A \sqsubseteq \exists r$	$\forall x\, A(x) \rightarrow \exists y\; r(x, y)$
funct(r)	$\forall x, y, z(r(x, y) \land r(x, z)) \rightarrow y = z$
$A \sqcap B \sqsubseteq C$,	$\forall x\, A(x) \land B(x) \rightarrow C(x)$
$\exists r.A \sqsubseteq B$	$\forall x\; \exists y\; r(x, y) \land A(y) \rightarrow B(x)$
$A \sqsubseteq \exists r.B$	$\forall x\, A(x) \rightarrow \exists y\; (r(x, y) \land B(y))$
…	…

Ngữ nghĩa dựa trên FOL

Ngữ nghĩa của biểu thức DL có thể được xác định dựa trên biểu thức tương ứng trong FOL.

- τ_R(r, x, y): biến đổi r(x, y) với hai biến tự do x, y
- τ_C(C, x): biến đổi C(x) với 1 biến tự do x
- Các biến đổi được thực hiện đệ quy

Viết lại đáy:

$$\tau_{\mathbf{R}}(u, x_i, x_j) = \mathbf{true}$$

$$\tau_{\mathbf{R}}(r, x_i, x_j) = r(x_i, x_j)$$

$$\tau_{\mathbf{R}}(r^-, x_i, x_j) = r(x_j, x_i)$$

$$\tau_{\mathbf{C}}(A, x_i) = A(x_i)$$

$$\tau_{\mathbf{C}}(\top, x_i) = \mathbf{true}$$

$$\tau_{\mathbf{C}}(\bot, x_i) = \mathbf{false}$$

$$\tau_{\mathbf{C}}(\{a_1, \ldots, a_n\}, x_i) = \bigvee_{j=1}^{n} x_i = a_j$$

Ngữ nghĩa dựa trên FOL$_{(2)}$

Các khái niệm phức tạp

$$\tau_{\mathbf{C}}(C \sqcap D, x_i) = \tau_{\mathbf{C}}(C, x_i) \wedge \tau_{\mathbf{C}}(D, x_i)$$

$$\tau_{\mathbf{C}}(C \sqcup D, x_i) = \tau_{\mathbf{C}}(C, x_i) \vee \tau_{\mathbf{C}}(D, x_i)$$

$$\tau_{\mathbf{C}}(\neg C, x_i) = \neg \tau_{\mathbf{C}}(C, x_i)$$

$$\tau_{\mathbf{C}}(\exists r.C, x_i) = \exists x_{i+1}.(\tau_{\mathbf{R}}(r, x_i, x_{i+1}) \wedge \tau_{\mathbf{C}}(C, x_{i+1}))$$

$$\tau_{\mathbf{C}}(\forall r.C, x_i) = \forall x_{i+1}.(\tau_{\mathbf{R}}(r, x_i, x_{i+1}) \rightarrow \tau_{\mathbf{C}}(C, x_{i+1}))$$

$$\tau_{\mathbf{C}}(\exists r.\mathsf{Self}, x_i) = \tau_{\mathbf{R}}(r, x_i, x_i)$$

$$\tau_{\mathbf{C}}(\geq nr.C, x_i) = \exists x_{i+1} \ldots x_{i+n}.\Big(\bigwedge_{j=i+1}^{i+n} \bigwedge_{k=j+1}^{i+n} (x_j \neq x_k) \wedge \bigwedge_{j=i+1}^{i+n} \bigwedge_{k=j+1}^{i+n} (\tau_{\mathbf{R}}(r, x_i, x_j) \wedge \tau_{\mathbf{C}}(C, x_j))$$

$$\tau_{\mathbf{C}}(\leq nr.C, x_i) = \neg \tau_{\mathbf{C}}(\geq (n+1)r.C, x_i)$$

Ngữ nghĩa dựa trên FOL$_{(3)}$

Các mệnh đề

$$\tau(C \sqsubseteq D) = \forall x_0(\tau_{\mathbf{C}}(C, x_0) \rightarrow \tau_{\mathbf{C}}(D, x_0))$$

$$\tau(r_1 \circ \ldots \circ r_n \sqsubseteq r) = \forall x_0 \ldots x_n(\bigwedge_{i=1}^{n} \tau_{\mathbf{R}}(r_i, x_{i-1}, x_i)) \rightarrow \tau_{\mathbf{R}}(r, x_0, x_n)$$

$$\tau(Dis(r, r')) = \forall x_0, x_1(\tau_{\mathbf{R}}(r, x_0, x_1) \rightarrow \neg\tau_{\mathbf{R}}(r', x_0, x_1)$$

$$\tau(Ref(r, r')) = \forall x\, \tau_{\mathbf{R}}(r, x, x)$$

$$\tau(Asym(r)) = \forall x_0, x_1.(\tau_{\mathbf{R}}(r, x_0, x_1) \rightarrow \neg\tau_{\mathbf{R}}(r, x_1, x_0))$$

Các kiểm tra

$$\tau(C(a)) = \tau_{\mathbf{C}}(C, x_0)[x_0/a]$$

$$\tau(r(a, b)) = \tau_{\mathbf{R}}(r, x_0, x_1)[x_0/a][x_1/b]$$

$$\tau(\neg r(a, b)) = \neg\tau(r(a, b))$$

$$\tau(a \approx b) = a = b$$

$$\tau(a \not\approx b) = \neg(a = b)$$

Các mở rộng của $\mathcal{ALC}$

Các ký hiệu

$((\mathcal{ALC} \mid \mathcal{S})[\mathcal{H}] \mid \mathcal{SR})[\mathcal{O}][\mathcal{I}][\mathcal{F} \mid \mathcal{N} \mid \mathcal{Q}]$

$\mathcal{ALC}$ và các mở rộng

- $\mathcal{S}$ - $\mathcal{ALC}$ + tính bắc cầu của vai trò
- $\mathcal{H}$ - vai trò con, tạo thành cấu trúc cây vai trò
- $\mathcal{O}$ - các lớp đóng
- $\mathcal{I}$ - các vai trò nghịch
- $\mathcal{N}$ ($\mathcal{Q}$) - giới hạn số lượng tùy ý (định kiểu)
- $\mathcal{F}$ - vai trò hàm
- $\mathcal{D}$ - kiểu dữ liệu
- $\mathcal{R}$ - mệnh đề bao gồm vai trò khái quát (rộng hơn $\mathcal{S}$)

Ghi chú:

$\mathcal{F}$ có thể được diễn đạt qua $\mathcal{N}$; $\mathcal{N}$ (và $\mathcal{F}$) có thể được diễn đạt qua $\mathcal{Q}$

Cú pháp DL: Khái quát

Các khái niệm		
$\mathcal{ALC}$	Nguyên tố Phủ định Và Hoặc Tồn tại Tất cả	A, B $\neg C$ $C \sqcap D$ $C \sqcup D$ $\exists r.C$ $\forall r.C$
$\mathcal{N}$ ($\mathcal{Q}$)	Tối thiểu Tối đa	$\geq n\ r$ ($\geq n\ r.C$) $\leq n\ r$ ($\leq n\ r.C$)
$\mathcal{O}$	Lớp đóng	$\{i_1, i_2, ..., i_n\}$
$\mathcal{R}$	Bản thân	$\exists r.Self$

Các vai trò		
$\mathcal{I}$	Nguyên tố Ngược	r r^-

Cú pháp DL: Khái quát (2)

Mệnh đề khái niệm (TBox)	
Lớp con Tương đương	$C \sqsubseteq D$ $C \equiv D$

Các mệnh đề vai trò (RBox)		
$\mathcal{H}$	Vai trò con	$r \sqsubseteq s$
$\mathcal{S}$	Bắc cầu	Trans(s)
$\mathcal{SR}$	Chuỗi vai trò, Vai trò không giao nhau	$r \circ r' \sqsubseteq s$ Dis(s, r)

Các mệnh đề khẳng định (ABox)	
Thành viên	C(a)
Vai trò	r(a, b)
Giống nhau	$a \approx b$
Khác nhau	$a \neq b$

Ontology (= CSTT)

- Tính bắc cầu và tính tách biệt là các đặc điểm của vai trò
- Trong $\mathcal{SROIQ}$ còn có tính chất bất đối xứng Asym(r), và phản xạ, Ref(r)

Các phiên bản OWL và lô-gic mô tả

OWL Full	Rộng hơn lô-gic mô tả
OWL DL	$\mathcal{SHOIN}$
OWL Lite	$\mathcal{SHIF}$
OWL 2 Full	Rộng hơn lô-gic mô tả
OWL 2 DL	$\mathcal{SROIQ}$
OWL 2 EL	$\mathcal{EL}^{++}$
OWL 2 QL	DL-Lite
OWL 2 RL	DLP (Description Logic Program)

Các khái niệm tương đương

$C \equiv D$, các biểu thức khái niệm C và D là tương đương nếu với tất cả các biểu diễn $\mathcal{I}$ thì $C^{\mathcal{I}} = D^{\mathcal{I}}$.

- Các tính chất giao hoán, kết hợp, lũy đẳng

$$\begin{aligned} C \sqcap D &\equiv D \sqcap C \\ (C \sqcap D) \sqcap E &\equiv C \sqcap (D \sqcap E) \\ C \sqcap C &\equiv C \end{aligned}$$

$$\begin{aligned} C \sqcup D &\equiv D \sqcup C \\ (C \sqcup D) \sqcup E &\equiv C \sqcup (D \sqcup E) \\ C \sqcup C &\equiv C \end{aligned}$$

- Phủ định kép

$$\neg\neg C \equiv C$$

- Phần bù và luật de Morgan

$$\begin{aligned} \neg\top &\equiv \bot \\ C \sqcap \neg C &\equiv \bot \\ \neg(C \sqcap D) &\equiv \neg D \sqcup \neg C \end{aligned}$$

$$\begin{aligned} \neg\bot &\equiv \top \\ C \sqcup \neg C &\equiv \top \\ \neg(C \sqcup D) &\equiv \neg D \sqcap \neg C \end{aligned}$$

Các khái niệm tương đương$_{(2)}$

- Phân phối, sáp nhập

$$
\begin{aligned}
(C \sqcup D) \sqcap E &\equiv (C \sqcap E) \sqcup (D \sqcap E) \\
(C \sqcap D) \sqcup E &\equiv (C \sqcup E) \sqcap (D \sqcup E) \\
C \sqcup (C \sqcap D) &\equiv C
\end{aligned}
\qquad
\begin{aligned}
(C \sqcup D) \sqcap C &\equiv C \\
(C \sqcap D) \sqcup C &\equiv C \\
C \sqcup (C \sqcap D) &\equiv C
\end{aligned}
$$

- Định lượng và cơ số

$$
\begin{aligned}
\neg \exists r.C &\equiv \forall r.\neg C \\
\neg \forall r.C &\equiv \exists r.\neg C \\
\neg \leq nr.C &\equiv \geq (n+1)r.C \\
\neg \geq (n+1)r.C &\equiv \leq nr.C
\end{aligned}
\qquad
\begin{aligned}
\geq 0r.C &\equiv \top \\
\geq 1r.C &\equiv \exists r.C \\
\leq 0r.C &\equiv \forall r.\neg C
\end{aligned}
$$

Tương đương mệnh đề và CSTT

- Tương đương Lloyd-Topor

$$\{A \sqcup B \sqsubseteq C\} \iff \{A \sqsubseteq C, B \sqsubseteq C\}$$
$$\{A \sqsubseteq B \sqcap C\} \iff \{A \sqsubseteq B, A \sqsubseteq C\}$$

- Biến đổi GCIs thành các mô tả khái niệm tổng quát

$$C \sqsubseteq D \iff \top \sqsubseteq \neg C \sqcup D$$

- Chuyển đổi giữa Abox và TBox

$$C(a) \iff \{a\} \sqsubseteq C$$
$$r(a,b) \iff \{a\} \sqsubseteq \exists r.\{b\}$$
$$\neg r(a,b) \iff \{a\} \sqsubseteq \neg\exists r.\{b\}$$
$$a \approx b \iff \{a\} \sqsubseteq \{b\}$$
$$a \not\approx b \iff \{a\} \sqsubseteq \neg\{b\}$$

Bao hàm khái niệm

$C \sqsubseteq D$, biểu thức khái niệm C nằm trong biểu thức khái niệm D nếu $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$ đúng với tất cả các biểu diễn $\mathcal{I}$.

Một số tính chất đơn giản:

- $C \sqsubseteq D \Leftrightarrow D \equiv C \sqcup D$
- $C \equiv D \Leftrightarrow C \sqsubseteq D$ và $D \sqsubseteq C$
- $C \sqsubseteq D \land D \sqsubseteq E \Rightarrow C \sqsubseteq E$
- $C \sqsubseteq D \Leftrightarrow \neg D \sqsubseteq \neg C$
- $C \sqsubseteq D \Rightarrow C \sqcap E \sqsubseteq D$
- $C \equiv D \Rightarrow C \sqcap E \equiv D \sqcap E$

Ví dụ 6.4. CSTT

$Human \sqsubseteq Animal \sqcap Biped$

$Man \equiv Human \sqcap Male$

$Male \sqsubseteq \neg Female$

$\{President\ Obama\} \equiv \{Barack\_Obama\}$

$\{john\} \sqsubseteq \neg\{peter\}$

$hasDaughter \sqsubseteq hasChild$

$hasChild \equiv hasParent^{-}$

$cost \equiv price$

Trans(ancestor)

Func(hasMother)

$Func(hasSSN^{-})$

Nội dung

6.1. Lô-gic bậc nhất

6.2. Lô-gic mô tả

6.3. Ngữ nghĩa lô-gic của OWL

6.4. Suy diễn tự động với OWL

Các cấu trúc khái niệm DL vs. lớp OWL

- ⊤ tương đương với owl:Thing
- ⊥ tương đương với owl:Nothing
- ⊓ tương đương với owl:intersectionOf
- ⊔ tương đương với owl:unionOf
- ¬ tương đương với owl:complementOf
- ∀ tương đương với owl:allValuesFrom
- ∃ tương đương với owl:someValuesFrom

Các tính năng tương tự trong DLs và OWL

OWL	**DL**
Lớp-class	khái niệm-concept
thuộc tính-property	vai trò-role
thuộc tính đối tượng	vai trò trừu tượng
object property	abstract role
thuộc tính dữ liệu	vai trò cụ thể
data property	concrete role
lớp đóng-oneOf	đơn vị-nominal
ontology	CSTT-knowledge base

Vai trò ngược

- Vai trò ngược được mô tả bằng owl:inverseOf
- Ngữ nghĩa của vai trò ngược (cặp thuận/nghịch) được định nghĩa như sau:
  - Chế tạo/được chế tạo bởi

$$(r^{-})^{\mathcal{I}} = \{(y, x) \mid (x, y) \in r^{\mathcal{I}}\}$$

Mở rộng $\mathcal{ALC}$ với vai trò ngược được ký hiệu là $\mathcal{I}$

Tính bắc cầu của vai trò

- Tính bắc cầu được mô tả bởi owl:TransitiveProperty
- Mệnh đề bắc cầu đối với 1 vai trò r có dạng Trans(r)
- Trans(r) đúng trong biểu diễn $\mathcal{I}$ nếu (x, y) ∈ $r^{\mathcal{I}}$ và (y, z) ∈ $r^{\mathcal{I}}$ thì (x, z) ∈ $r^{\mathcal{I}}$, và ký hiệu là $\mathcal{I} \models$Trans(r)

Mở rộng $\mathcal{ALC}$ với tính chất bắc cầu được ký hiệu là $\mathcal{S}$.

Tính chất hàm của vai trò

- Tính chất hàm được mô tả với owl:FunctionalProperty
- Mệnh đề hàm đối với vai trò r có dạng Func(r)
- Func(r) đúng trong một biểu diễn $\mathcal{I}$ nếu $(x, y_1) \in r^{\mathcal{I}}$ và $(x, y_2) \in r^{\mathcal{I}}$ thì $y_1 = y_2$, ký hiệu là $\mathcal{I} \models \mathrm{Func}(r)$

Mở rộng $\mathcal{ALC}$ với tính chất hàm được ký hiệu là $\mathcal{F}$

Giới hạn số lượng (tự do)

- Các giới hạn cơ số tùy ý được mô tả bằng owl:maxCardinality, owl:minCardinality, và owl:cardinality
- Đối với một vai trò đơn giản s và một số nguyên n, ≤ n s, ≥ n s, và = n s là các khái niệm
- ngữ nghĩa được định nghĩa như sau:

$$(\leq ns)^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} \mid \#\{y \in \Delta^{\mathcal{I}} \mid (x, y) \in s^{\mathcal{I}}\} \leq n\}$$

$$(\geq ns)^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} \mid \#\{y \in \Delta^{\mathcal{I}} \mid (x, y) \in s^{\mathcal{I}}\} \geq n\}$$

$$(= ns)^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} \mid \#\{y \in \Delta^{\mathcal{I}} \mid (x, y) \in s^{\mathcal{I}}\} = n\}$$

Mở rộng $\mathcal{ALC}$ với giới hạn số lượng tự do được ký hiệu là $\mathcal{N}$

Ví dụ 6.5. Các giới hạn trong OWL

- owl:hasValue bắt buộc sử dụng 1 phần tử cụ thể

```
:Woman owl:equivalentClass [
    a owl:Restriction;
    owl:onProperty :hasGender;
    owl:hasValue :female
] .
```

- Trong lô-gic mô tả:

$$Woman \equiv \exists hasGender.\{female\}$$

Giới hạn số lượng tự do trong FOL

- FOL yêu cầu cân bằng hoặc các đại lượng đếm
- Biến đổi được thực hiện như sau (tương tự như cho π_y):

$$\pi_x (\leq n\ s) = \exists^{\leq n} y.(s(x, y))$$
$$\pi_x (\geq n\ s) = \exists^{\geq n} y.(s(x, y))$$
$$\pi_x (= n\ s) = \exists^{\leq n} y.(s(x, y)) \land \exists^{\geq n} y.(s(x, y))$$

- Các biểu thức sau là tương đương:

$\neg(\leq n\ s) = \geq n + 1\ s$ $\quad$ $\neg(> n\ s) = \leq n - 1\ s,\ n \geq 1$

$\neg(\geq 0\ s) = \bot$ $\quad$ $\geq 1\ s = \exists s.\top$

$\leq 0\ s = \forall s.\bot$ $\quad$ $\top \sqsubseteq \leq 1 s = \mathrm{Func}(s)$

Các lớp đóng

- Được mô tả bởi owl:oneOf
- Định nghĩa lớp bằng cách liệt kê tất cả các phần tử của nó
- với $a_1, \ldots, a_n$ là các thành phần, trong DL $\{a_1, \ldots, a_n\}$ là 1 khái niệm liệt kê.
- Ngữ nghĩa được định nghĩa như sau:

$$\text{DL: } (\{a_1, \ldots, a_n\})^{\mathcal{I}} = \{a^{\mathcal{I}}_1, \ldots, a^{\mathcal{I}}_n\}$$

$$\text{FOL: } \pi_x(\{a_1, \ldots, a_n\}) = (x = a_1 \vee \ldots \vee x = a_n)$$

Mở rộng $\mathcal{ALC}$ với khái niệm liệt kê được ký hiệu là $\mathcal{O}$

Biến đổi DL thành OWL

- Tiếp nối các phần lô-gic, OWL cung cấp thành phần mô tả ontology

$$[\![\mathcal{KB}]\!] = \mathsf{Pre} + \mathsf{Dec}(\mathcal{KB}) + \Sigma_{\alpha \in \mathcal{KB}} [\![\alpha]\!]$$

$$\mathsf{Pre} = \begin{cases} \texttt{@prefix owl: <http://www.w3.org/2002/07/owl\#>} \\ \texttt{@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema\#>} \\ \texttt{@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns\#>} \\ \texttt{@prefix xsd: <http://www.w3.org/2001/XMLSchema\#>} \end{cases}$$

$$\begin{aligned} \mathsf{Dec}(\mathcal{KB}) &= \Sigma_{A \in \mathsf{N}_C(\mathcal{KB})}\, A \ \texttt{rdf:type owl:Class} \\ &+ \Sigma_{r \in \mathsf{N}_R(\mathcal{KB})}\, r \ \texttt{rdf:type owl:ObjectProperty} \end{aligned}$$

Chuyển đổi giữa DL và OWL

$$
\begin{aligned}
r_1 \circ \ldots \circ r_n \sqsubseteq r &\doteq r \ \texttt{owl:propertyChainAxiom} \ (r_1, \ldots, r_n) \ . \\
\mathsf{Dis}(r, r') &\doteq r \ \texttt{owl:propertyDisjointWith} \ r' \ . \\
C \sqsubseteq D &\doteq C \ \texttt{owl:subClassOf} \ D \ . \\
C(a) &\doteq a \ \texttt{rdf:type} \ C \ . \\
r(a, b) &\doteq a \ r \ b \ . \\
r^{-}(a, b) &\doteq b \ r \ a \ . \\
\neg r(a, b) &\doteq [] \ \texttt{rdf:type owl:NegativePropertyAssertion} \ ; \\
&\qquad \texttt{owl:assertionProperty} \ r \ ; \\
&\qquad \texttt{owl:sourceIndividual} \ a \ ; \ \texttt{owl:targetValue} \ b \ . \\
a \approx b &\doteq a \ \texttt{owl:sameAs} \ b \ . \\
a \not\approx b &\doteq a \ \texttt{owl:differentFrom} \ b \ .
\end{aligned}
$$

Chuyển đổi giữa DL và OWL$_{(2)}$

$$
\begin{aligned}
u &\doteq \texttt{owl:topObjectProperty} \;. \\
r &\doteq r \;. \\
r^{-} &\doteq [\ \texttt{owl:inverseOf}\ \ {:}r\ ]\,. \\
A &\doteq A\,. \\
\top &\doteq \texttt{owl:Thing} \;. \\
\bot &\doteq \texttt{owl:Nothing} \;. \\
\{a_1,\ldots,a_n\} &\doteq [\ \texttt{rdf:type owl:Class ; owl:oneOf}\ ({:}a_1 \ldots {:}a_n)\ ]\,. \\
\neg C &\doteq [\ \texttt{rdf:type owl:Class ; owl:complementOf}\ C\ ]\,. \\
C_1 \sqcap \ldots \sqcap C_n &\doteq [\ \texttt{rdf:type owl:Class ; owl:intersectionOf}\ (C_1 \ldots C_n)\ ]\,. \\
C_1 \sqcup \ldots \sqcup C_n &\doteq [\ \texttt{rdf:type owl:Class ; owl:unionOf}\ (C_1 \ldots C_n)\ ]\,.
\end{aligned}
$$

Chuyển đổi giữa DL và OWL$_{(3)}$

$$\exists r.C \doteq [\ \texttt{rdf:type owl:Restriction}\ ;$$
$$\texttt{owl:onProperty}\ r\ ;\ \texttt{owl:someValuesFrom}\ C\ ]\ .$$

$$\forall r.C \doteq [\ \texttt{rdf:type owl:Restriction}\ ;$$
$$\texttt{owl:onProperty}\ r\ ;\ \texttt{owl:allValuesFrom}\ C\ ]\ .$$

$$\exists r.\mathsf{Self} \doteq [\ \texttt{rdf:type owl:Restriction}\ ;$$
$$\texttt{owl:onProperty}\ r\ ;\ \texttt{owl:hasSelf ''true'' xsd:boolean}\ ]\ .$$

$$\geq rn.C \doteq [\ \texttt{rdf:type owl:Restriction}\ ;$$
$$\texttt{owl:minQualifiedCardinality}\ n\ \texttt{xsd:nonNegativeInteger;}$$
$$\texttt{owl:onProperty}\ r\ ;\ \texttt{owl:onClass}\ C\ ]\ .$$

$$\leq rn.C \doteq [\ \texttt{rdf:type owl:Restriction}\ ;$$
$$\texttt{owl:maxQualifiedCardinality}\ n\ \texttt{xsd:nonNegativeInteger;}$$
$$\texttt{owl:onProperty}\ r\ ;\ \texttt{owl:onClass}\ C\ ]\ .$$

$\exists r.\top \sqsubseteq C$ $\quad r$ `rdfs:domain` C .

$\top \sqsubseteq \forall r.C$ $\quad r$ `rdfs:range` C .

Ví dụ 6.6. CSTT DL & OWL

RBox $\mathcal{R}$

$\text{own} \sqsubseteq \text{careFor}$

"Nếu ai đó sở hữu cái gì thì họ quan tâm về nó"

TBox $\mathcal{T}$

$\text{Healthy} \sqsubseteq \neg\text{Dead}$

$\text{Cat} \sqsubseteq \text{Dead} \sqcup \text{Alive}$

$\text{HappyCatOwner} \sqsubseteq \exists \text{owns.Cat} \sqcap \forall \text{caresFor.Healthy}$

"Sống khỏe mạnh là không chết"; " Mèo có thể đã chết hoặc còn sống"; "Chủ nhân hạnh phúc của mèo là người sở hữu mèo và tất cả mèo mà người đó quan tâm đều khỏe mạnh."

ABox $\mathcal{A}$

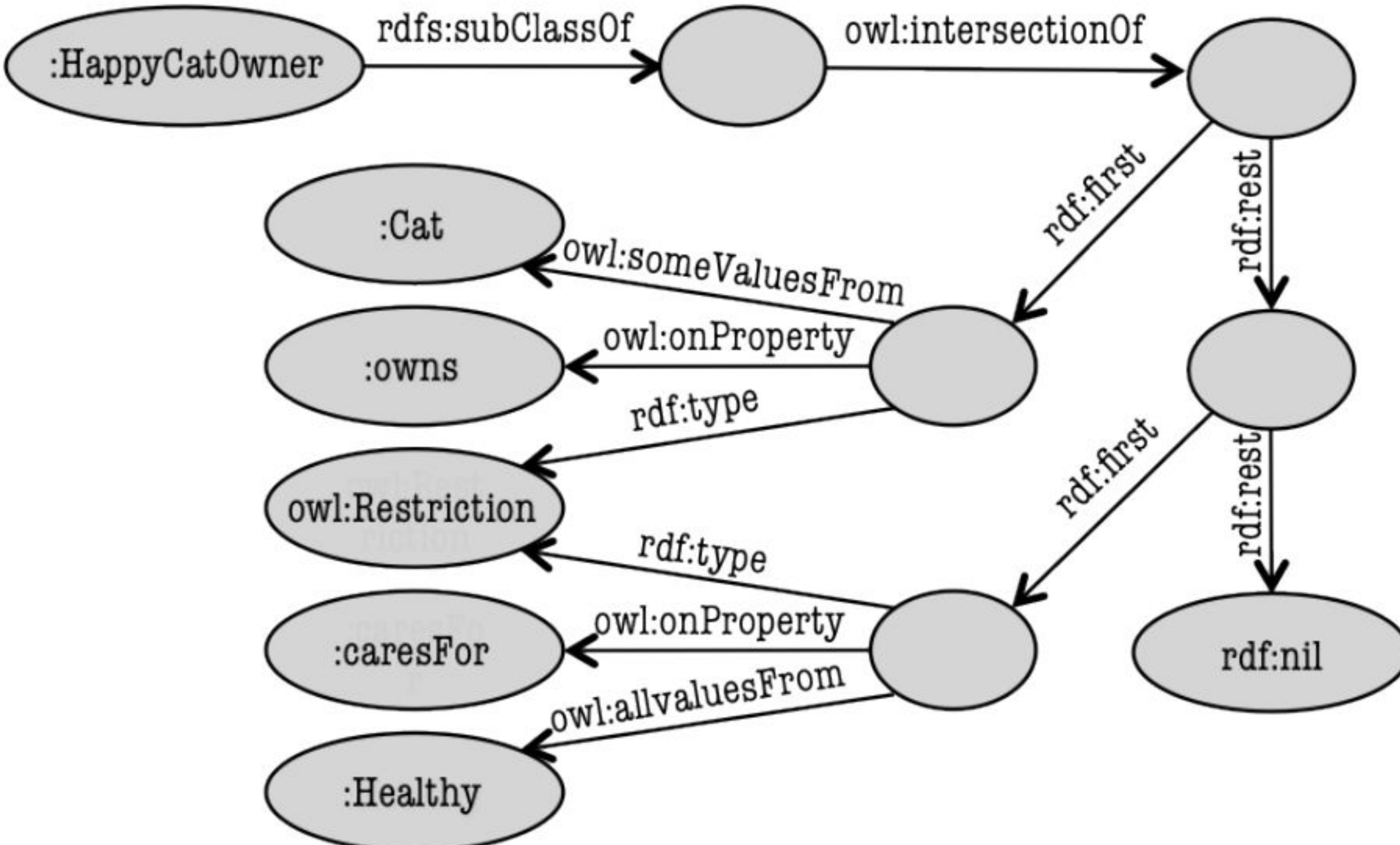
HappyCatOwner(NVA)

"NVA là người sở hữu mèo hạnh phúc."

Ví dụ 6.6. CSTT DL & OWL$_{(2)}$

```
:owns rdfs:SubPropertyOf :caresFor .
:Healthy rdfs:subClassOf [ owl:complementOf :Dead ].
:Cat rdfs:subClassOf [ owl:unionOf (:Dead :Alive) ] .
:HappyCatOwner rdfs:subClassOf
          [ owl:intersectionOf
          ( [ rdf:type owl:restriction ;
              owl:onProperty :owns ;
              owl:someValuesFrom :Cat ]
            [ rdf:type owl:Restriction ;
              owl:onProperty :caresFor ;
              owl:allValuesFrom :Healthy ] ) ] .
:NVA rdf:type :HappyCatOwner .
```

Ví dụ 6.6. CSTT DL & OWL$_{(3)}$



Biến đổi các giả thuyết ABox

Nếu các lớp đóng được hỗ trợ thì ABox có thể được chuyển đổi thành (1 phần) TBox

$C(a) = \{a\} \sqsubseteq C$

$r(a, b) = \{a\} \sqsubseteq \exists r.\{b\}$

$\neg r(a, b) = \{a\} \sqsubseteq \forall r.(\neg\{b\})$

$a \approx b = \{a\} \equiv \{b\}$

$a \not\approx b = \{a\} \sqsubseteq \neg\{b\}$

Nội dung

6.1. Lô-gic bậc nhất

6.2. Lô-gic mô tả

6.3. Ngữ nghĩa lô-gic của OWL

6.4. Suy diễn tự động với OWL

Giả thuyết thế giới mở vs thế giới đóng

Giả thuyết thế giới mở - Open World Assumption (OWA)

- Có thể tồn tại các thành phần mới nếu nó không bị bác bỏ một cách tường minh
- OWL sử dụng OWA

Giả thuyết thế giới đóng - Closed World Assumption (CWA)

- Được cho rằng CSTT có tất cả các phần tử và dữ kiện

Ví dụ 6.7. OWA và CWA

	Có phải tất cả con của Bill đều là con trai?	Không rõ nếu chúng ta cho rằng không biết tất cả về Bill	Nếu chúng ta biết tất cả thì tất cả con của Bill đều là con trai
child(bill, bob) Man(bob)	⊨? $(\forall$child.Man)(bill)	Trả lời theo DL không rõ	Prolog Yes
(≤ 1 child)(bill)	⊨? $(\forall$child.Man)(bill)	Yes	Yes

Các vấn đề suy diễn

Suy diễn lô-gic trong CSTT

- Đặt $\mathcal{I}$ là 1 biểu diễn, $\mathcal{T}$ là 1 TBox, $\mathcal{A}$ là 1 ABox và $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ là 1 CSTT
- $\mathcal{I}$ là mô hình của $\mathcal{T}$ nếu $\mathcal{I} \models$ ax cho tất cả mệnh đề trong $\mathcal{T}$, viết là $\mathcal{I} \models \mathcal{T}$.
- $\mathcal{I}$ là mô hình của $\mathcal{A}$, nếu $\mathcal{I} \models$ ax cho tất cả mệnh đề trong $\mathcal{A}$, viết là $\mathcal{I} \models \mathcal{A}$.
- $\mathcal{I}$ là mô hình của $\mathcal{K}$, nếu $\mathcal{I} \models \mathcal{T}$ và $\mathcal{I} \models \mathcal{A}$
- Một mệnh đề ax được suy ra từ $\mathcal{K}$, viết là $\mathcal{K} \models$ ax, nếu tất cả mô hình $\mathcal{I}$ của $\mathcal{K}$ cũng là mô hình của ax.

Các vấn đề suy diễn cho CSTT $\mathcal{K}$

- Tính nhất quán toàn cục của CSTT:
  - Giả sử $\mathcal{K} \models^? false$, $\mathcal{K} \models^? \top \sqsubseteq \bot$?
  - CSTT có hợp lý hay không?
- Tính nhất quán lớp: $\mathcal{K} \models^? C \sqsubseteq \bot$?
  - Lớp C có phải là lớp rỗng ko?
- Lớp bao gồm: $\mathcal{K} \models^? C \sqsubseteq D$?
  - Cấu trúc cây của CSTT
- Lớp tương đương: $\mathcal{K} \models^? C \equiv D$?
  - Hai lớp có tập thành phần giống nhau hay không

Các vấn đề suy diễn cho CSTT $\mathcal{K}_{(2)}$

- Lớp không giao nhau: $\mathcal{K} \vDash^{?} C \sqcap D \sqsubseteq \bot$?
  - Hai lớp có giao nhau hay không?
- Phần tử của lớp: $\mathcal{K} \vDash^{?} C(a)$?
  - Phần tử a có nằm trong lớp C hay không?
- Truy xuất phần tử: Tìm tất cả x sao cho $\mathcal{K} \vDash C(x)$
  - Tìm tất cả các phần tử (đã biết) của lớp C

Các dạng kiểm tra ABox

Một kiểm tra ABox có thể có một trong các dạng sau:

- C(a) (kiểm tra khái niệm)
- r(a, b) (kiểm tra vai trò)
- ¬r(a, b) (kiểm tra vai trò phủ định)
- a ≈ b (kiểm tra tương đương)
- a ≉ b (kiểm tra khác biệt)

Tính khả quyết của OWL DL

- OWL DL là 1 phần của FOL, như vậy có thể sử dụng tiến trình suy diễn FOL (phân giải Tableaux)
  - Nhưng không đảm bảo tính dừng
- Vấn đề: Tìm kiếm các giải thuật có đảm bảo tính dừng
- Không có giải pháp đơn giản cho vấn đề này.

Suy diễn trong OWL 2

- OWL 2 mở rộng các tính năng đã giới thiệu với các cấu trúc mới
- OWL 2 cũng định nghĩa các thành phần đơn giản hơn
- Nhiều công cụ cho phép suy diễn tự động
- Trình biên soạn hỗ trợ tạo ontologies/CSTT

Bài tập 6.1

Xây dựng các Ontology với OWL DL và Lô-gic mô tả.

- The class `Vegetable` is a subclass of `PizzaTopping`.
- The class `PizzaTopping` does not share any elements with the class `Pizza`.
- The individual `aubergine` is an element of the class `Vegetable`.
- The abstract role `hasTopping` is only used for relationships between elements of the classes `Pizza` and `PizzaTopping`.
- The class `VegPizza` consists of those elements which are in the class `NoMeatPizza` and in the class `NoFishPizza`.
- The role `hasTopping` is a subrole of `hasIngredient`.

Bài tập 6.2

Chuyển đổi sang OWL

$$\mathtt{Human} \sqsubseteq \exists \mathtt{hasMother}.\mathtt{Human}$$

$$\exists \mathtt{hasMother}.(\exists \mathtt{hasMother}.\mathtt{Human}) \sqsubseteq \mathtt{Grandchild}$$

$$\mathtt{Human}(\mathtt{anupriyaAnkolekar})$$

Bài tập 6.3.

```
:Person a owl:Class.
:Food a owl:Class.
:eats rdfs:domain :Person;
        rdfs:range :Food.
:Maverick :eats :Steak.
:Vegetarian a owl:Class;
                 rdfs:subClassOf :Person.
:VegetarianFood a owl:Class;
                 rdfs:subClassOf :Food.
:Jen a :Vegetarian;
        :eats :Marzipan.
```

a) Hãy bổ xung các mô tả dựa trên owl cho lớp :VegetarianFood để có thể suy ra :Marzipan là :VegetarianFood, nhưng :Steak thì không?

b) Chuyển sang DL.

Bài tập 6.4

1. Tạo Ontology trong phạm vi ngữ nghĩa owl, sử dụng cú pháp Turtle dựa trên các mô tả:

a. Bus Drivers are Drivers who drive buses.

b. Drivers are people who drive at least one vehicle.

c. Bus is a subclass of vehicle

Sử dụng các từ sau cho các lớp: BusDriver, Driver, Bus, Vehicle

2. Chuyển Ontology thu được ở 1 sang DL.

3. Hãy tìm 3 bộ 3 có thể suy diễn được từ các dữ kiện

  B001 rdf:type Bus.

  NVA drive B001.

dựa trên Ontology đã tạo trong 1?