

representação de conhecimento - raciocínio

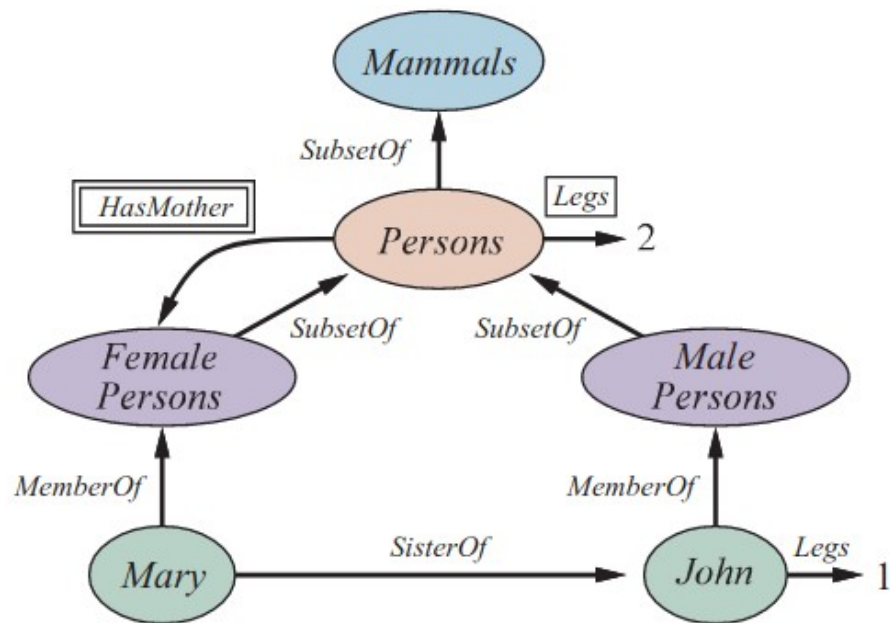
categorias

são elementos de composição de esquemas de representação de conhecimento de larga escala

dois tipos de sistemas principais

- redes semânticas na realidade, também uma forma de lógica
representação gráfica e algoritmos de inferência eficientes
- lógicas descritivas
linguagem formal e algoritmos eficientes

redes semânticas



☐ $\forall x, x \in Persons \Rightarrow$
 $[\forall y HasMother(x, y) \Rightarrow$
 $y \in FemalePersons]$

☐ $\forall x, x \in Persons \Rightarrow Legs(x, 2)$

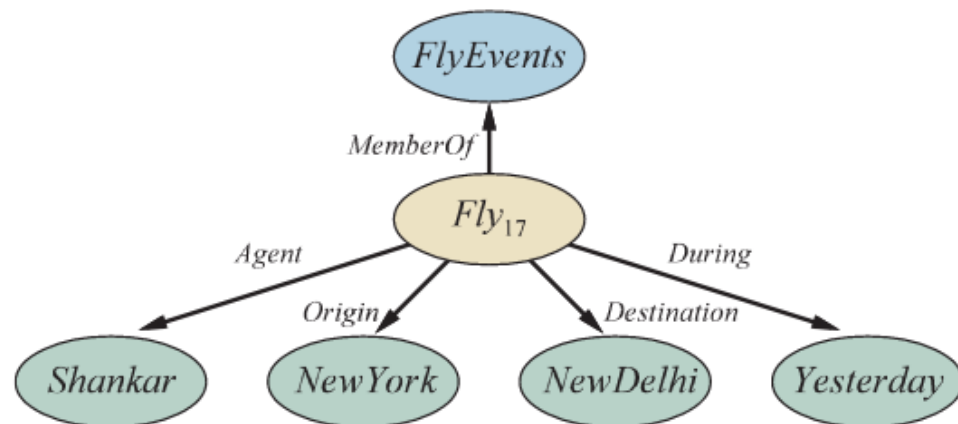
- relações de herança
- permite fazer inferências seguindo as setas
Mary tem duas pernas
- representa conhecimento padrão
John tem 1 perna – substitui padrão

relações binárias

rede semântica está limitada a este tipo de relações

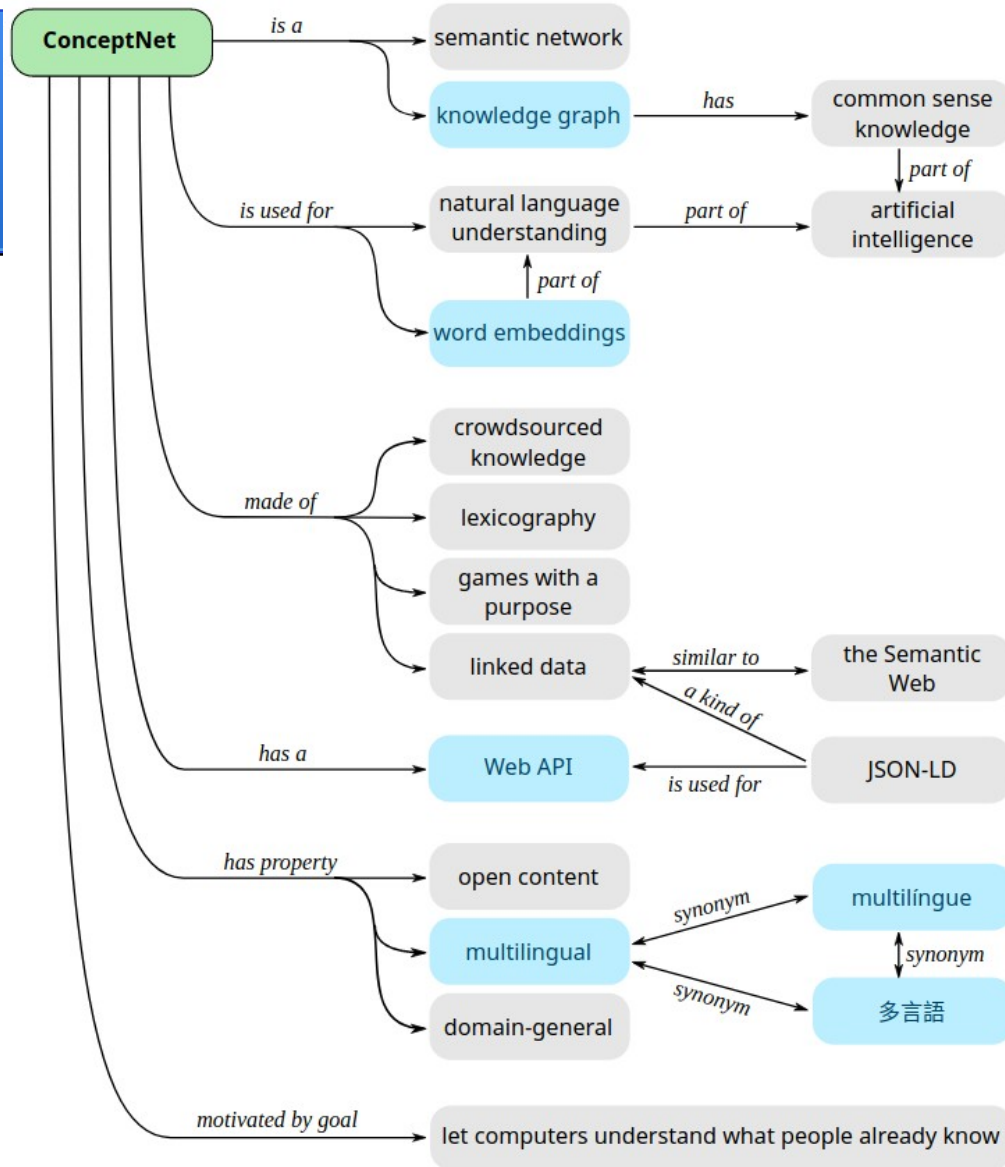
o que força a ontologias ricas (em relações) onde conceitos são objetificados

ex:



Fly(Shankar, NewYork, NewDelhi, Yesterday)

ConceptNet



lógicas de descrição

criadas para facilitar a descrição de
definições e propriedades de categorias

inferências

subsunção: verificação de uma categoria ser subconjunto de outra, verificando as respetivas propriedades

classificação: verificação se um objeto pertence a uma categoria
(em algumas) **consistência:** satisfação lógica dos critérios de pertença

linguagem CLASSIC

Concept → **Thing** | *ConceptName*

| **And**(*Concept*,...)

| **All**(*RoleName*, *Concept*)

| **AtLeast**(*Integer*, *RoleName*)

| **AtMost**(*Integer*, *RoleName*)

| **Fills**(*RoleName*, *IndividualName*,...)

| **SameAs**(*Path*, *Path*)

| **OneOf**(*IndividualName*,...)

Path → [*RoleName*,...]

ConceptName → *Adult* | *Female* | *Male* | ...

RoleName → *Spouse* | *Daughter* | *Son* | ...

não permite uma asserção de uma categoria ser um subconjunto de outra

essa relação tem de ser derivável de aspetos das descrições das categorias

relações de disjunção

ex. em CLASSIC

definição de universidade

Universidade = And(Instit_Ensino, Nível_Sup, Três_ciclos)

em lógica de 1ª ordem

Universidade(x) = Instit_Ensino(x) \wedge Nível_Sup(x) \wedge Três_ciclos(x)

outro exemplo

And(Man, AtLeast(3, Son), AtMost(2, Daughter),

tentar descrever
em LPO!

All(Son, And(Unemployed, Married, All(Spouse, Doctor)))

All(Daughter, And(Professor, Fills(Department, Physics, Math))))

caraterísticas de DLs

inferência tratável

teste de subsunção resolvido em tempo polinomial pode haver exceções
o que limita o poder de descrever problemas complexos

DLs em geral não têm

negação

disjunção

→ CLASSIC tem de forma limitada
em *Fills* e *OneOf*

raciocínio com informação típica

ou *default*

dois tipos principais de lógica

- lógica padrão
- lógica de circunscrição

lógica padrão

regras padrão, ex:

$Pássaro(x) : \underline{Voa(x)} / \underline{Voa(x)}$

se $Pássaro(x)$ é verdade e se $\underline{Voa(x)}$ é consistente com a base de conhecimento, a conclusão padrão é $\underline{Voa(x)}$

em geral uma regra padrão é

$P : J_1, \dots, J_n / C$ qualquer variável que apareça em J_i ou C , tem de aparecer em P

P – pré-requisito, C – conclusão, J_i – justificações
se alguma é falsa,
não se pode tirar a conclusão

contradições por herança múltipla

ex:

amigos dos animais protegem os animais

carnívoros não protegem os animais

Fulano é amigo dos animais e come animais

$AmigoAnimais(Fulano) \wedge Carnívoro(Fulano)$

$AmigoAnimais(X) : ProtegeAnimais(X) / ProtegeAnimais(X)$

$Carnívoro(X) : \neg ProtegeAnimais(X) / \neg ProtegeAnimais(X)$

há duas extensões
uma em que *Fulano*
é amigo dos animais
e outra em que é
carnívoro

lógica de circunscrição

o voo dos pássaros

$$Pássaro(x) \wedge \neg Anormal_1(x) \Rightarrow Voa(x)$$

raciocinador circunscritivo assume $\neg Anormal_1(x)$ a não ser que se verifique $Anormal_1(x)$

lógica que prefere o modelo da *KB* que tiver menos objetos anormais
no caso de dois modelos preferidos contraditórios (extensões na lógica padrão, exemplo da pg. anterior) mantém-se agnóstica

sistemas de manutenção da verdade

revisão de crenças

conhecimento padrão que, em certos casos, tem exceções mediante novo conhecimento específico

≠ atualização de crenças
conhecimento específico que se alterou

KB com asserção P (ex. resultante de uma inferência)

e queremos fazer a asserção $TELL(KB, \neg P)$

primeiro temos de fazer $RETRACT(KB, P)$ **hélas!!!**

revisão de crenças

quando a afirmação a retirar já foi usada para inferir novas afirmações adicionadas à KB...

suponhamos que $P \Rightarrow Q$

não se pode simplesmente retirar Q , porque pode também verificar-se R e $R \Rightarrow Q$

sistemas de revisão de crenças, *truth maintenance systems* (TMS) tratam destes problemas

TMS

manter nº de ordem das asserções feitas à KB

P_1 a P_n

e quando se faz $\text{RETRACT}(KB, P_i)$

reverte-se ao estado $i-1$, removendo todas as asserções até P_i
depois fazem-se novamente as inferências possíveis desde $i-1$

mas é impraticável para sistemas muito ativos

JTMS

TMS baseado em justificações

cada afirmação na KB é anotada com uma justificação
que consiste no conjunto de afirmações de que foi inferida

as retrações são eficientes

o tempo necessário para retrair P depende apenas no nº de
afirmações inferidas a partir de P

ex: a KB tem $P \Rightarrow Q$, então $\text{TELL}(P)$ faz adicionar Q

com a justificação $\{P, P \Rightarrow Q\}$

JTMS - retração

tendo Q com a justificação $\{P, P \Rightarrow Q\}$

se é executado $\text{RETRACT}(P)$, então Q também é removido

e se a justificação de Q for $\{P, P \vee R \Rightarrow Q\}$, também!

mas se a justificação de Q for $\{R, P \vee R \Rightarrow Q\}$ mantém-se

os JTMS não apagam logo a afirmação, marcam-na “fora” da KB

se depois uma asserção a restaurar, marcam-na “dentro”

retêm as cadeias de inferência sem terem de rederivar conhecimento

ATMS – *assumption based* TMS

facilita mudanças de contexto entre mundos hipotéticos
mantém registo, para cada afirmação das **assunções** que a tornariam verdadeira

TMSs em geral têm mecanismos para gerar **explicações**

explicação de P é um conjunto de afirmações $E : E \models P$

mas também podem incluir **assunções**
afirmações que não são conhecidas como verdadeiras, mas que, se fossem, seriam suficientes para provar P

TMS complexidade

os TMS têm complexidade NP-hard
ainda assim são úteis
porque, em muitos casos práticos,
permitem aumentar a eficiência
da inferência em KB