Artificial Intelligence

More Knowledge Representation and Reasoning (Approaches Other Than Logic)

Nilsson - Chapter 18

some taxonomic knowledge

"all office machines get their energy from wall outlets"

"all printers are office machines"

"all laser printers are printers"

"Hobbes is a laser printer."

production systems (= rule-based systems)

resolution

$$\frac{\text{P OR Q, NOT Q OR R}}{\text{P OR R}}$$

modus ponens

$$\frac{\text{P, P IMPLIES Q}}{\text{Q}}$$

$$\frac{\text{P(a), FORALL x P(x) IMPLIES Q(x)}}{\text{Q(a)}}$$

production systems (= rule-based systems)

rule memory:

if OfficeMachine(x) then EnergySource(x, WallOutlet)
if IsPrinter(x) then IsOfficeMachine(x)
if LaserPrinter(x) then IsPrinter(x)

could have general actions here, such as "delete from working memory"
"print to screen" etc.

working memory:

LaserPrinter(Hobbes)
Instructor(Sven)
HasLegs(Sven,2)

production systems (= rule-based systems)

sometimes

if "a person has jaundice"
then it is highly likely that "their eyes are yellow"

production systems (= rule-based systems)

"backward chaining"

When a new query q is asked
  if a matching fact q' is known, return the unifier
  for each rule whose consequent q' matches q
    attempt to prove each premise of the rule by backward chaining

backward chaining is query-driven (= hypothesis-driven)

production systems (= rule-based systems)

"forward chaining"

When a new fact p is added to the knowledge base
  for each rule such that p unifies with a premise
    if the other premises are known
    then add the conclusion to the knowledge base and continue chaining

forward chaining is data-driven

production systems (= rule-based systems)

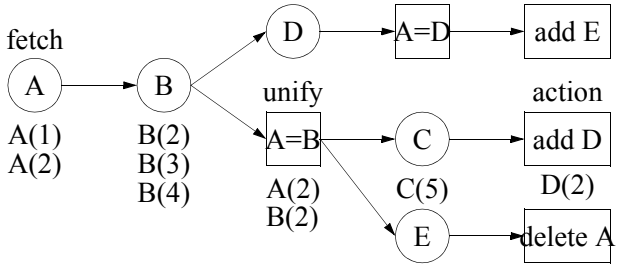
- match phase
- conflict resolution phase
  - don't fire again with same bindings
  - recency
  - specificity
    if Mammal(x) then add Legs(x,4)
    if Mammal(x) and Human(x) then add Legs(x,2)
  - priority
    if ControlPanel(p) and Dusty(p) then Action(Dust(p))
    if ControlPanel(p) and WarningLightOn(p) then Action(Evacuate)
- act phase

production systems (= rule-based systems)

making matching efficient with rete networks
(avoids duplication of effort between different rules and over time)

if A(x) and B(x) and C(y) then add D(x)
if A(x) and B(y) and D(x) then add E(x)
if A(x) and B(x) and E(x) then delete A(x)

A(1), A(2), B(2), B(3), B(4), C(5)



production systems (= rule-based systems)

- modularity
- control isolated from knowledge
- easy modification
- able to give explanations for its conclusions

widely used, for example for configuration and design

some taxonomic knowledge

“all office machines get their energy from wall outlets”

“all printers are office machines”

“all laser printers are printers”

“Hobbes is a laser printer.”

first-order logic

FORALL x IsOfficeMachine(x) IMPLIES EnergySource(x,WallOutlet)
= “all office machines get their energy from wall outlets”

FORALL x IsPrinter(x) IMPLIES IsOfficeMachine(x)
= “all printers are office machines”

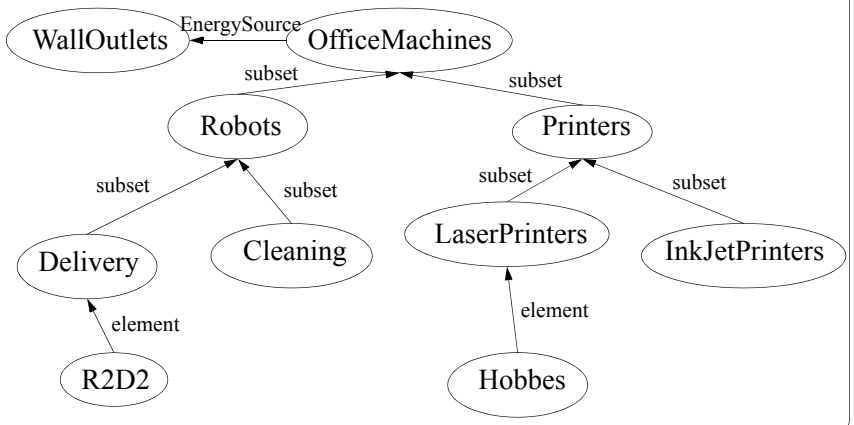
FORALL x IsLaserPrinter(x) IMPLIES IsPrinter(x)
= “all laser printers are printers”

IsLaserPrinter(Hobbes)
= “Hobbes is a laser printer.”

IsPrinter(Hobbes)
EnergySource(Hobbes, WallOutlet)

semantic networks

(good for classification and taxonomies)



semantic networks

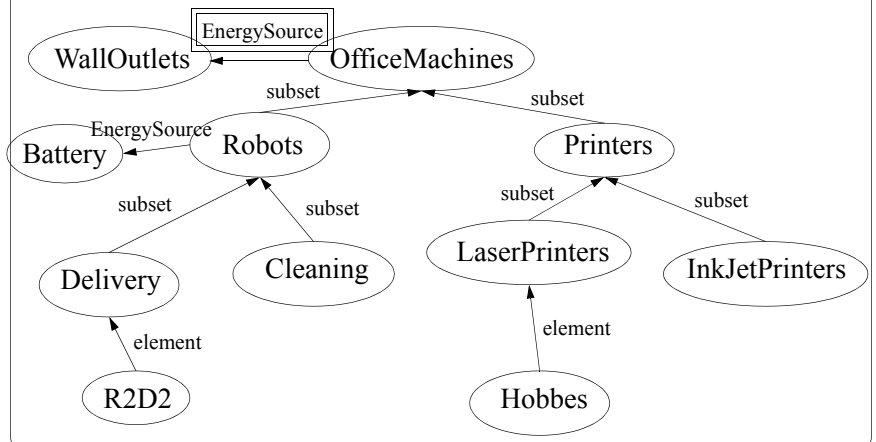
link type			semantics
A	—subset→	B	FOR x A(x) IMPLIES B(x)
A	—element→	B	B(A)
A	—R→	B	R(A,B)
A	—[R]→	B	FORALL x A(x) IMPLIES R(x,B)
A	—[R]→	B	FORALL x EXISTS y A(x) IMPLIES B(y) AND R(x,y)

semantic networks

link type			example		
A	—subset→	B	Cats	—subset→	Mammals
A	—element→	B	Bill	—element→	Cats
A	—R→	B	Bill	—Age→	12
A	—[R]→	B	Birds	—[Legs]→	2
A	—[R]→	B	Birds	—[Parent]→	Birds

default reasoning with semantic networks

default reasoning is nonmonotonic
and thus cannot be done with first-order logic



default reasoning with semantic networks

default reasoning is nonmonotonic
and thus cannot be done with first-order logic

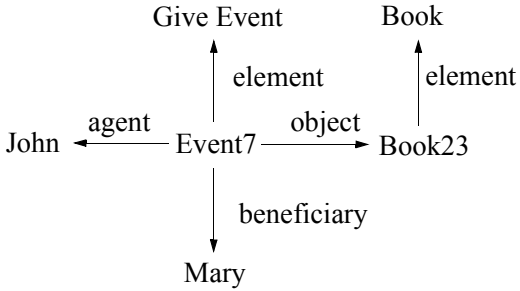
Assume that KB is a subset of KB' and KB entails A then
KB' also entails A

for example:
if IsBird(Tweety) entails CanFly(Tweety) then
IsBird(Tweety) AND IsPenguin(Tweety) entails CanFly(Tweety)

learning additional facts does not change conclusions in first-order logic
(first-order logic is monotonic)

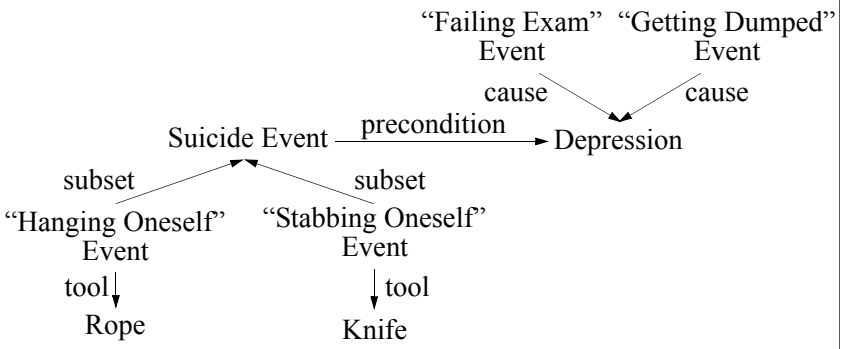
semantic networks

John gave a book to Mary.



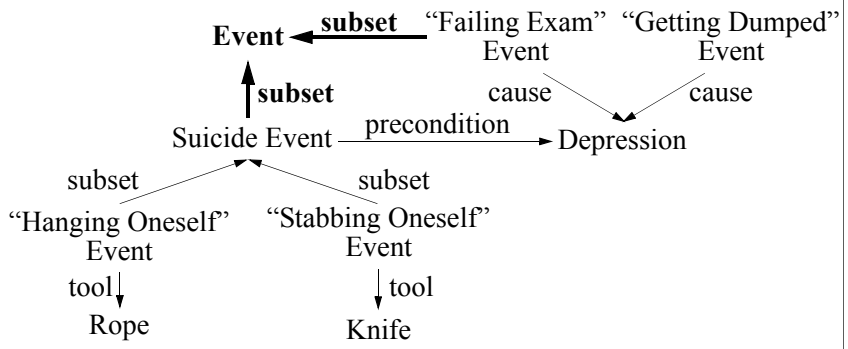
spreading activation (marker passing) for story understanding

John failed the exam. He reached for a rope.



spreading activation (marker passing) for story understanding

John failed the exam. He reached for a rope.



spreading activation (marker passing) for word disambiguation

John bought a jaguar and immediately got a speeding ticket.

frames

Printers

subsetof:	OfficeMachines
supersetof:	{LaserPrinters, InkJetPrinters}
energysource:	WallOutlet
creator:	SvenKoenig
date:	Feb 23, 2005

Hobbes

elementof:	LaserPrinters
energyconsumption:	180 Watt/h
energycostperhour:	utility:energycost * this:energyconsumption
creator:	SvenKoenig
date:	Feb 23, 2005

frame systems and semantic networks compared to first-order logic

- more efficient (can use special purpose procedures)
- easier to read and understand by humans
- semantics is often not well defined
- easier to implement (can use special purpose procedures)
- problems with multiple inheritance of incompatible properties

- less expressive (for example, negation and disjunctions are problems)
- more expressive (inheritance with exceptions, procedural attachements)