Artificial Intelligence
Knowledge Representation

Dr. Ahmed Mateen

Consider the following set of statements prove that **Marcus is dead**

1. Marcus was a man
2. Marcus was a Pompeian
3. Marcus was born in 40 A.D.
4. All men are mortal
5. All Pompeians died when volcano erupted in 79 A.D.
6. No mortal lies longer than 150 years
7. It is now 1991 A.D.

Solution 1

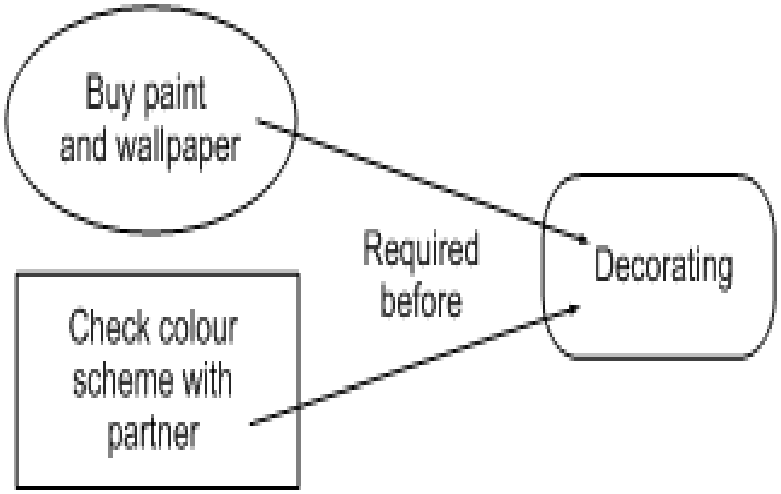
1 Marcus was a man	axiom 1
4 All men are mortal	axiom 4
8 Marcus is mortal	1,4
3 Marcus was born in 40 A.D.	axiom 3
7 It is now 1991 A.D.	axiom 7
9 Marcus' age is 1951 years	3,7
6 No mortal lives longer than 150 years	axiom 6
10 Marcus is dead	8,6,9

Solution 2

7	It is now 1991 A.D.	Axiom 7
5	All Pompeians died in 79 A.D.	Axiom 5
11	All Pompeians are dead now	7,5
2	Marcus was a Pompeian	axiom 2
12	Marcus is dead	11,2

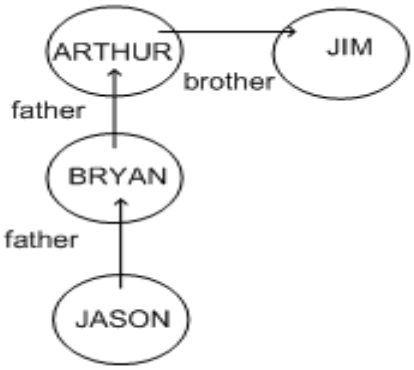
Graphical Representation

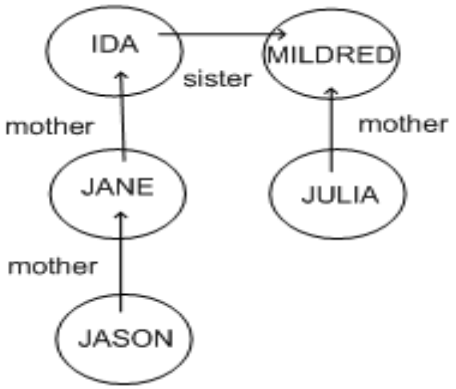
- Humans draw diagrams all the time, e.g.
  - Causal relationships



Graphical Representation

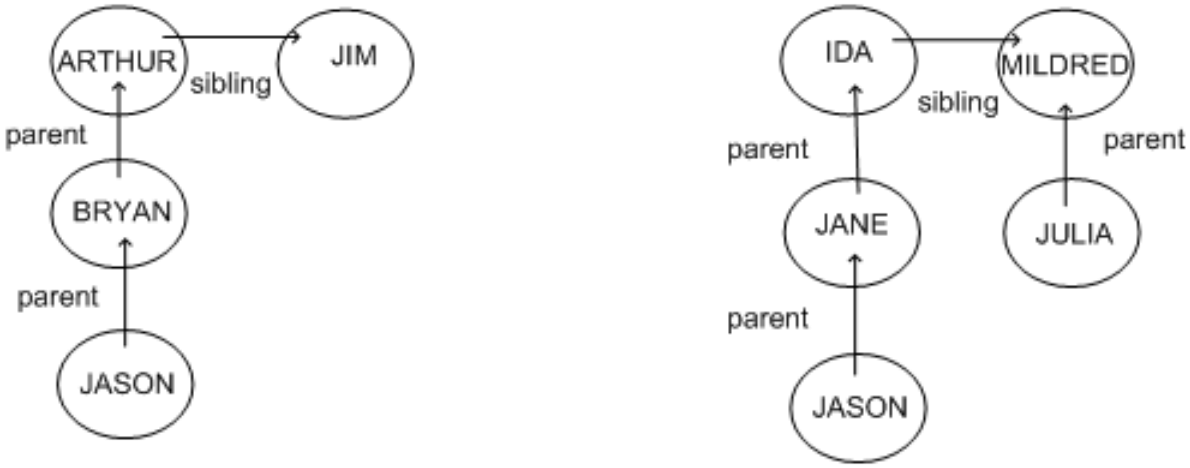
- Graphs easy to store in a computer
- To be of any use must impose a formalism





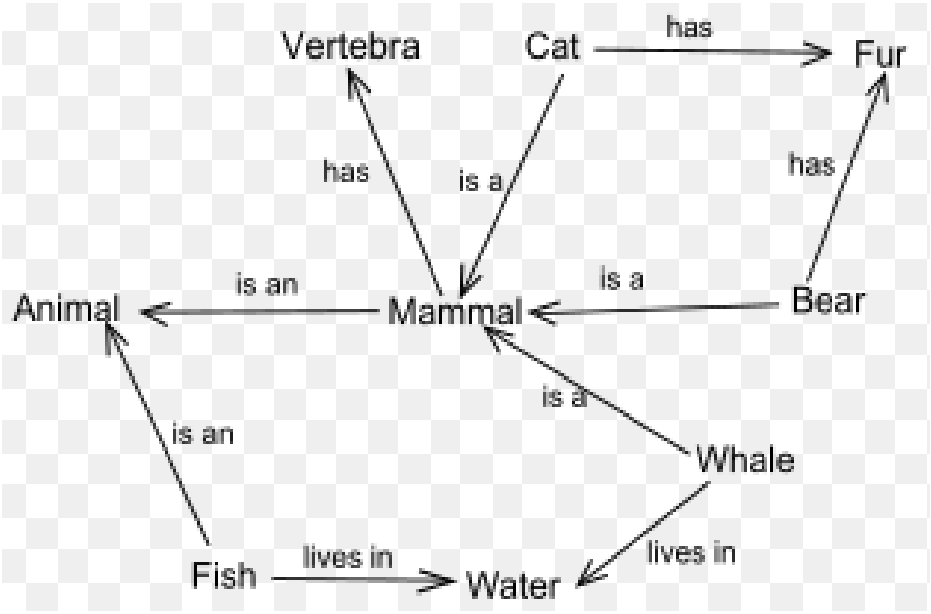
  - Jason is 15, Bryan is 40, Arthur is 70, Jim is 74
  - How old is Julia?

Semantic networks



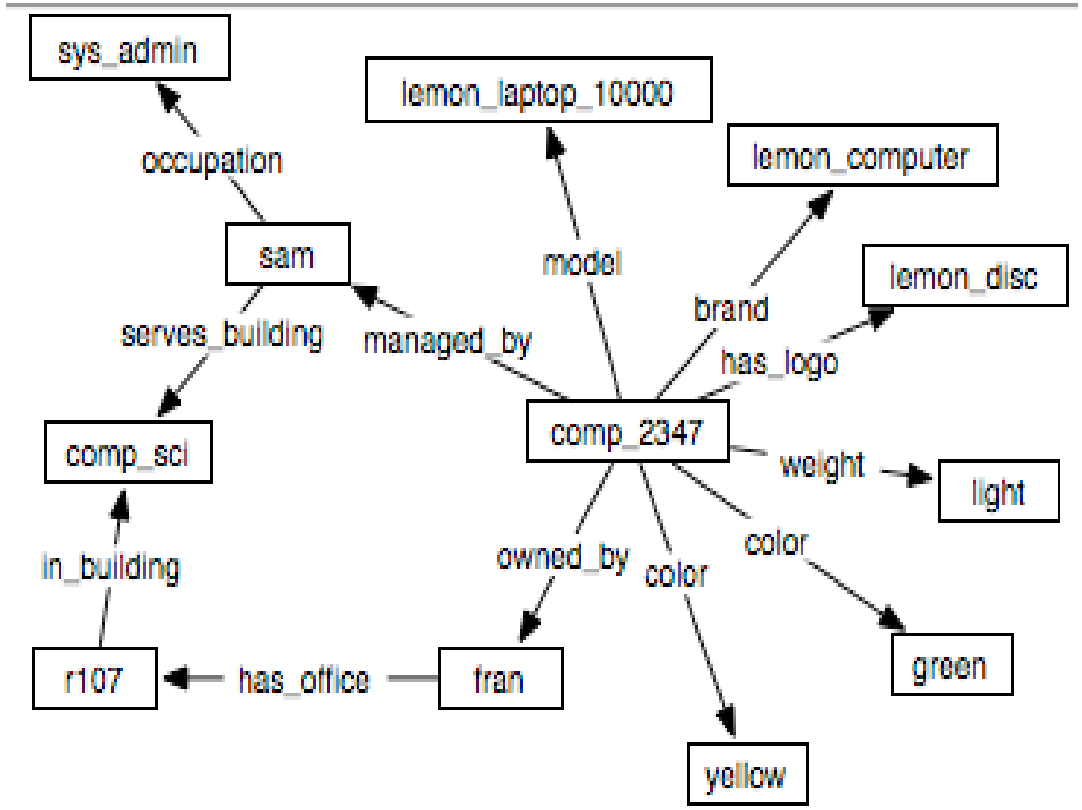
- Because the syntax is the same
  - We can guess that Julia's age is similar to Bryan's

Semantic Network



Semantic Networks

- Graphical representation (a graph)
  - Links indicate subset, member, relation, ...
- Equivalent to logical statements (usually FOL)
  - Easier to understand than FOL?
  - Specialised SN reasoning algorithms can be faster
- Example: natural language understanding
  - Sentences with same meaning have same graphs


sys_admin
lemon_laptop_10000
occupation
lemon_computer
sam
model
lemon_disc
brand
serves_building
managed_by
has_logo
comp_sci
comp_2347
weight
light
owned_by
color
color
in_building
green
r107
has_office
fran
yellow

interpret the *prop* relation in terms of a graph

prop(comp_2347 , owned_by , fran).
prop(comp_2347 , managed_by , sam).
prop(comp_2347 , model , lemon_laptop_10000).
prop(comp_2347 , brand , lemon_computer).
prop(comp_2347 , has_logo , lemon_disc).
prop(comp_2347 , color , green).
prop(comp_2347 , color , yellow).
prop(comp_2347 , weight , light).
prop(fran , has_office , r107).
prop(r107 , in_building , comp_sci).

Interpretation

- The network also shows how the knowledge is structured.
- For example, it is easy to see that,

Computer number *2347* is owned by someone (Fran) whose office (r107) is in the *comp_sci* building.

The direct indexing evident in the graph can be used by humans and machines.

Types of Relationship in Semantic networks

- There are many types of relationships that can be used in semantic networks. The following are four of them.
  - “is a”
  - “is an instance of”
  - “is a part of”
  - “has”

- The “is-a” relationship between class and superclass (Figure 1)

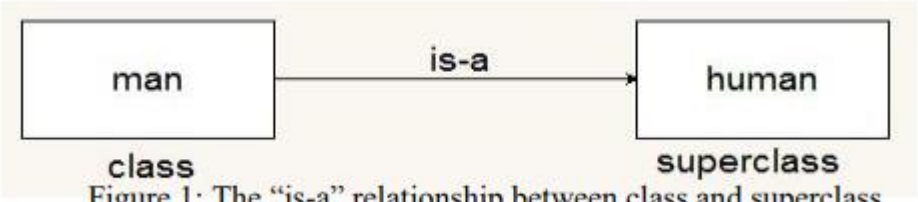


Figure 1: The “is-a” relationship between class and superclass

The “is an instance of“ relationship between instance and class (Figure 2);

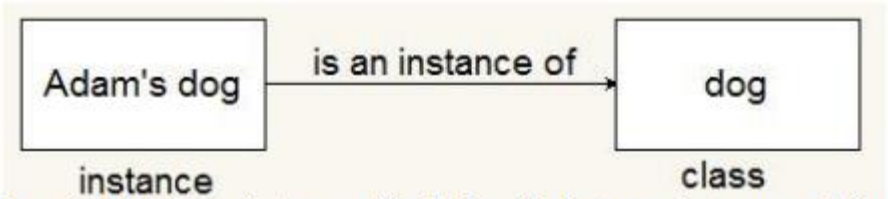


Figure 2: The “is an instance of“ relationship between instance and class

- The “is a part of“ relationship between part and whole (Figure 3);

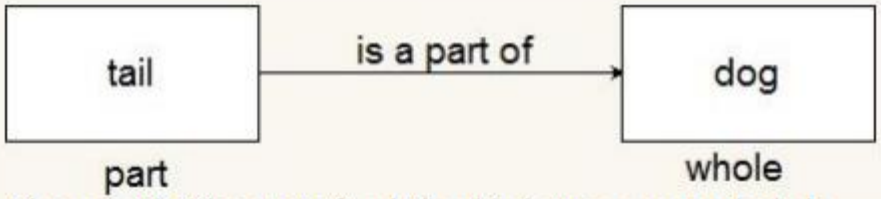


Figure 3: The “is a part of“ relationship between part and whole

- The “has” relationship between object and attribute (Figure 4).

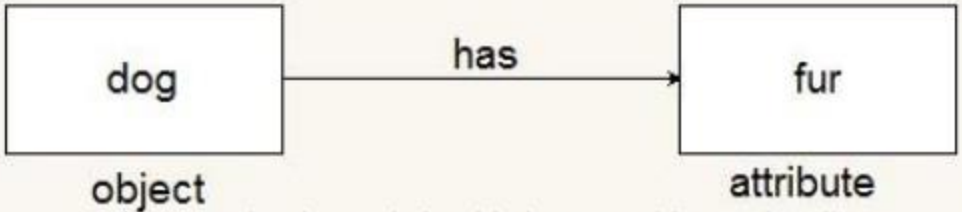
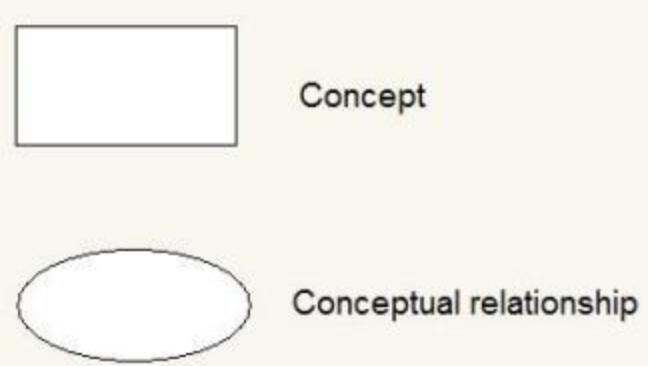


Figure 4: The “has” relationship between object and attribute

Conceptual Graphs

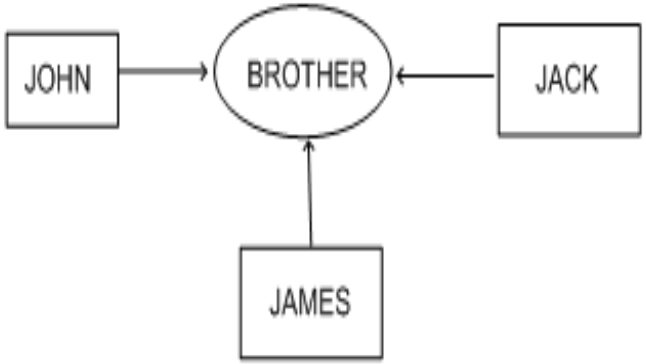
- There are two kinds of nodes that can be used in conceptual graph
  - “concept”
  - “conceptual relationship”



  - Conceptual graph arcs
    - No arc between two concepts or conceptual relationship

Conceptual Graphs

- Concept nodes can be
  - Concrete (visualisable) such as restaurant, name
  - Abstract (not easily visualisable) such as anger
- Edges do not have labels
  - Instead, conceptual relation nodes
  - Easy to represent relations between multiple objects



Semantic v/s Conceptual