Statistical Reasoning

Probability and Bayes' theorem and causal networks, reasoning belief network

Introduction

Suppose you are trying to determine if a patient has inhalational anthrax. You observe the following symptoms:

- The patient has a cough
- The patient has a fever
- The patient has difficulty breathing

Introduction

You would like to determine how likely the patient is infected with inhalational anthrax given that the patient has a cough, a fever, and difficulty breathing

We are not 100% certain that the patient has anthrax because of these symptoms. We are dealing with uncertainty!

Introduction

Now suppose you order an x-ray and observe that the patient has a wide mediastinum.

Your belief that that the patient is infected with inhalational anthrax is now much higher.

Introduction

- In the previous slides, what you observed affected your belief that the patient is infected with anthrax
- This is called reasoning with uncertainty
- Wouldn't it be nice if we had some methodology for reasoning with uncertainty? Why in fact, we do…

- How does these uncertainty come??

Sources of Uncertainty

- Uncertain **inputs --** missing and/or noisy data
- Uncertain **knowledge**
  - Multiple causes lead to multiple effects
  - Incomplete enumeration of conditions or effects
  - Incomplete knowledge of causality in the domain
  - Probabilistic/stochastic effects
- Uncertain **outputs**
  - Abduction and induction are inherently uncertain
  - Default reasoning, even deductive, is uncertain
  - Incomplete deductive inference may be uncertain

- Probabilistic reasoning only gives probabilistic results (summarizes uncertainty from various sources)

Decision making with uncertainty

Rational behavior:

- For each possible action, identify the possible outcomes
- Compute the **probability** of each outcome
- Compute the **utility** of each outcome
- Compute the probability-weighted **(expected) utility** over possible outcomes for each action
- Select action with the highest expected utility (principle of **Maximum Expected Utility**)

At a glance

- if we roll two dice, each showing one of six possible numbers, the number of total unique rolls is 6*6 = 36. We distinguish the dice in some way (a first and second or left and right die). Here is a listing of the joint possibilities for the dice:
  (1,1) (1,2) (1,3) (1,4) (1,5) (1,6)
  (2,1) (2,2) (2,3) (2,4) (2,5) (2,6)
  (3,1) (3,2) (3,3) (3,4) (3,5) (3,6)
  (4,1) (4,2) (4,3) (4,4) (4,5) (4,6)
  (5,1) (5,2) (5,3) (5,4) (5,5) (5,6)
  (6,1) (6,2) (6,3) (6,4) (6,5) (6,6)

- The number of rolls which add up to 4 is 3 ((1,3), (2,2), (3,1)), so the probability of rolling a total of 4 is 3/36 = 1/12.
- This does not mean 8.3% true, but 8.3% chance of it being true.

Probabilities anyway?

Kolmogorov showed that three simple axioms lead to the rules of probability theory

1. All probabilities are between 0 and 1:

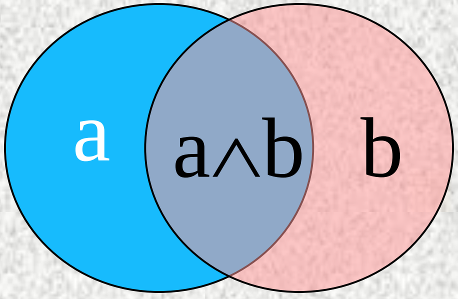
   $0 \leq P(a) \leq 1$

2. Valid propositions (tautologies) have probability 1, and unsatisfiable propositions have probability 0:

   $P(true) = 1 \; ; \; P(false) = 0$

3. The probability of a disjunction is given by:

   $P(a \vee b) = P(a) + P(b) - P(a \wedge b)$



Probability theory

- **Random variables**
  - Domain
- **Atomic event**: complete specification of state
- **Prior probability**: degree of belief without any other evidence
- **Joint probability**: matrix of combined probabilities of a set of variables

Probability theory

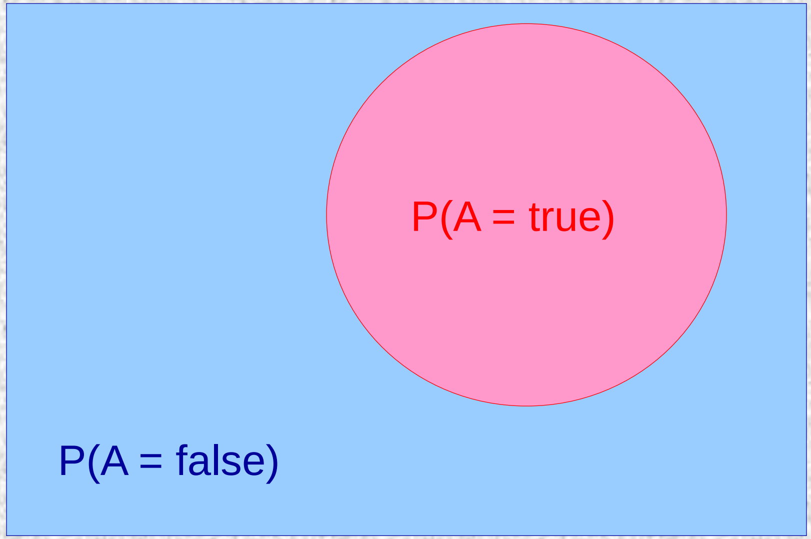
- **Conditional probability**: prob. of effect given causes
- **Computing conditional probs**:
  - $P(a \mid b) = P(a \wedge b) / P(b)$
  - P(b): **normalizing** constant
- **Product rule**:
  - $P(a \wedge b) = P(a \mid b) * P(b)$

Probabilities Theory

We will write *P(A = true)* to mean the probability that *A = true*.

What is probability? It is the relative frequency with which an outcome would be obtained if the process were repeated a large number of times under similar conditions*

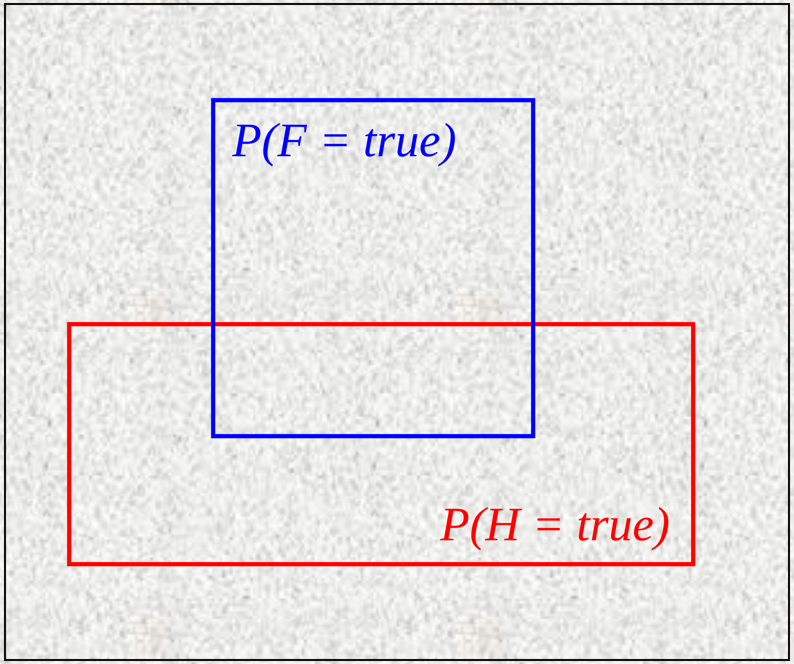
The sum of the red and blue areas is 1



*Ahem…there's also the Bayesian definition which says probability is your degree of belief in an outcome

Conditional Probability

- P(A = *true* | B = *true*) = Out of all the outcomes in which B is true, how many also have A equal to true
- Read this as: "Probability of A conditioned on B" or "Probability of A given B"



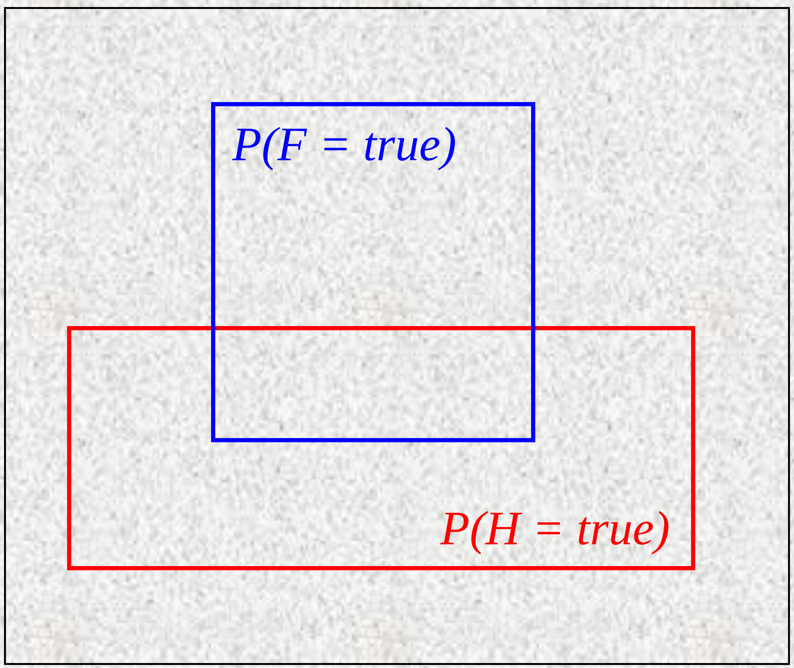
H = "Have a headache"
F = "Coming down with Flu"

P(H = *true*) = 1/10
P(F = *true*) = 1/40
P(H = *true* | F = *true*) = 1/2

"Headaches are rare and flu is rarer, but if you're coming down with flu there's a 50-50 chance you'll have a headache."

The Joint Probability Distribution

- We will write *P(A = true, B = true)* to mean "the probability of *A = true* **and** *B = true*"
- Notice that:



$$P(H=true|F=true)$$

$$= \frac{\text{Area of "H and F" region}}{\text{Area of "F" region}}$$

$$= \frac{P(H = true, F = true)}{P(F = true)}$$

In general, $P(X|Y)=P(X,Y)/P(Y)$

The Joint Probability Distribution

- Joint probabilities can be between any number of variables

  eg. *P(A = true, B = true, C = true)*
- For each combination of variables, we need to say how probable that combination is
- The probabilities of these combinations need to sum to 1

A	**B**	**C**	**P(A,B,C)**
false	false	false	0.1
false	false	true	0.2
false	true	false	0.05
false	true	true	0.05
true	false	false	0.3
true	false	true	0.1
true	true	false	0.05
true	true	true	0.15

Sums to 1

The Joint Probability Distribution

- Once you have the joint probability distribution, you can calculate any probability involving *A*, *B*, and *C*
- Note: May need to use marginalization and Bayes rule, (both of which are not discussed in these slides)

A	B	C	P(A,B,C)
false	false	false	0.1
false	false	true	0.2
false	true	false	0.05
false	true	true	0.05
true	false	false	0.3
true	false	true	0.1
true	true	false	0.05
true	true	true	0.15

Examples of things you can compute:

- $P(A=true)$ = sum of $P(A,B,C)$ in rows with $A=true$
- $P(A=true, B = true \mid C=true) =$

  $P(A = true, B = true, C = true) / P(C = true)$

The Problem with the Joint Distribution

- Lots of entries in the table to fill up!
- For *k* Boolean random variables, you need a table of size 2^k
- How do we use fewer numbers? Need the concept of independence

A	**B**	**C**	**P(A,B,C)**
false	false	false	0.1
false	false	true	0.2
false	true	false	0.05
false	true	true	0.05
true	false	false	0.3
true	false	true	0.1
true	true	false	0.05
true	true	true	0.15

Independence

Variables *A* and *B* are independent if any of the following hold:

- $P(A,B) = P(A)\,P(B)$
- $P(A \mid B) = P(A)$
- $P(B \mid A) = P(B)$

This says that knowing the outcome of *A* does not tell me anything new about the outcome of *B*.

Independence

How is independence useful?

- Suppose you have n coin flips and you want to calculate the joint distribution $\boldsymbol{P}(C_1, \ldots, C_n)$
- If the coin flips are not independent, you need 2^n values in the table
- If the coin flips are independent, then

$$P(C_1, \ldots, C_n) = \prod_{i=1}^{n} P(C_i)$$

Each P(C_i) table has 2 entries and there are n of them for a total of 2n values

Conditional Independence

Variables *A* and *B* are conditionally independent given *C* if any of the following hold:

- $P(A, B \mid C) = P(A \mid C)\, P(B \mid C)$
- $P(A \mid B, C) = P(A \mid C)$
- $P(B \mid A, C) = P(B \mid C)$

Knowing *C* tells me everything about *B*. I don’t gain anything by knowing *A* (either because *A* doesn’t influence *B* or because knowing *C* provides all the information knowing *A* would give)

Independence

- When sets of variables don't affect each others' probabilities, we call them **independent**, and can easily compute their joint and conditional probability:

$Independent(A, B) \rightarrow P(A \wedge B) = P(A) * P(B), \; P(A \mid B) = P(A)$

- {moonPhase, lightLevel} *might* be independent of {burglary, alarm, earthquake}
  - Maybe not: crooks may be more likely to burglarize houses during a new moon (and hence little light)
  - But if we know the light level, the moon phase doesn't affect whether we are burglarized
  - If burglarized, light level doesn't affect if alarm goes off
- Need a more complex notion of independence and methods for reasoning about the relationships

Axioms of Probability

- Bayes' Rule
  - Given a hypothesis (H) and evidence (E), and given that P(E) = 0, what is P(H|E)?
- Many times rules and information are uncertain, yet we still want to say something about the consequent; namely, the degree to which it can be believed. A British cleric and mathematician, Thomas Bayes, suggested an approach.
- Recall the two forms of the product rule:
  - P(ab) = P(a) * P(b|a)
  - P(ab) = P(b) * P(a|b)
- If we equate the two right-hand sides and divide by P(a), we get $$P(b \mid a) = \frac{P(a \mid b)P(b)}{P(a)}$$

Example

- Bayes' rule is useful when we have three of the four parts of the equation.
- For example, a doctor knows that meningitis causes a stiff neck in 50% of such cases. The prior probability of having meningitis is 1/50,000 and the prior probability of any patient having a stiff neck is 1/20.
- What is the probability that a patient has meningitis if they have a stiff neck?
- H = "Patient has meningitis“
- E = "Patient has stiff neck"

$$P(H|E) = \frac{P(E|H) * P(H)}{P(E)}$$

$$P(H|E) = (0.5*.00002) / .05 = .0002$$

Bayesian inference

- In the setting of diagnostic/evidential reasoning

$$\begin{array}{cccc} & H_i \quad P(H_i) & & \text{hypotheses} \\ & P(E_j \mid H_i) & & \\ E_1 & E_j & E_m & \text{evidence/manifestations} \end{array}$$

  – Know prior probability of hypothesis $P(H_i)$

  conditional probability $P(E_j \mid H_i)$

  – Want to compute the *posterior probability* $P(H_i \mid E_j)$

- Bayes' s theorem (formula 1):

$$P(H_i \mid E_j) = P(H_i) * P(E_j \mid H_i) / P(E_j)$$

Simple Bayesian diagnostic reasoning

- Also known as: Naive Bayes classifier
- Knowledge base:
  - Evidence / manifestations: $E_1, \ldots E_m$
  - Hypotheses / disorders: $H_1, \ldots H_n$

    Note: E_j and H_i are **binary**; hypotheses are **mutually exclusive** (non-overlapping) and **exhaustive** (cover all possible cases)
  - Conditional probabilities: $P(E_j \mid H_i)$, $i = 1, \ldots n$; $j = 1, \ldots m$
- Cases (evidence for a particular instance): $E_1, \ldots, E_l$
- Goal: Find the hypothesis H_i with the highest posterior
  - $\text{Max}_i \, P(H_i \mid E_1, \ldots, E_l)$

Simple Bayesian diagnostic reasoning

- Bayes' rule says that

  $P(H_i \mid E_1 \ldots E_m) = P(E_1 \ldots E_m \mid H_i)\ P(H_i) / P(E_1 \ldots E_m)$
- Assume each evidence E_i is conditionally independent of the others, *given* a hypothesis H_i, then:

  $P(E_1 \ldots E_m \mid H_i) = \prod^m_{j=1} P(E_j \mid H_i)$
- If we only care about relative probabilities for the H_i, then we have:

  $P(H_i \mid E_1 \ldots E_m) = \alpha\ P(H_i) \prod^m_{j=1} P(E_j \mid H_i)$

Limitations

- Cannot easily handle multi-fault situations, nor cases where intermediate (hidden) causes exist:
  - Disease D causes syndrome S, which causes correlated manifestations M_1 and M_2
- Consider a composite hypothesis $H_1 \wedge H_2$, where H_1 and H_2 are independent. What's the relative posterior?

$$
\begin{aligned}
P(H_1 \wedge H_2 \mid E_1, \ldots, E_l) &= \alpha\, P(E_1, \ldots, E_l \mid H_1 \wedge H_2)\, P(H_1 \wedge H_2) \\
&= \alpha\, P(E_1, \ldots, E_l \mid H_1 \wedge H_2)\, P(H_1)\, P(H_2) \\
&= \alpha \prod^{l}_{j=1} P(E_j \mid H_1 \wedge H_2)\, P(H_1)\, P(H_2)
\end{aligned}
$$

- How do we compute $P(E_j \mid H_1 \wedge H_2)$?

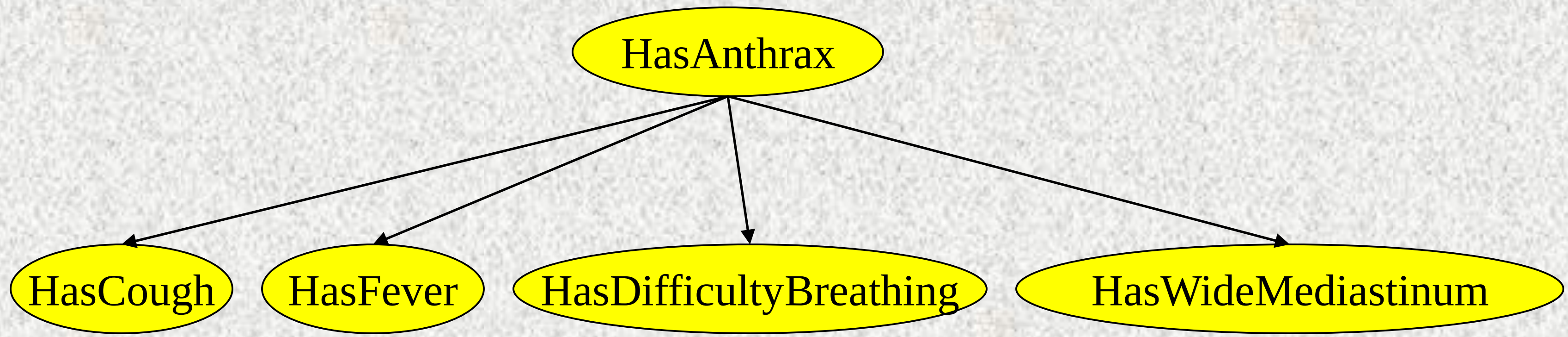
Limitations

- Assume H1 and H2 are independent, given E1, …, El?
  - $P(H_1 \wedge H_2 \mid E_1, \ldots, E_l) = P(H_1 \mid E_1, \ldots, E_l)\, P(H_2 \mid E_1, \ldots, E_l)$
- This is a very unreasonable assumption
  - Earthquake and Burglar are independent, but *not* given Alarm:
    - P(burglar | alarm, earthquake) << P(burglar | alarm)
- Another limitation is that simple application of Bayes's rule doesn't allow us to handle causal chaining:
  - A: this year's weather; B: cotton production; C: next year's cotton price
  - A influences C indirectly: A→ B → C
  - P(C | B, A) = P(C | B)
- Need a richer representation to model interacting hypotheses, conditional independence, and causal chaining
- Next: conditional independence and Bayesian networks!

Summary

- Probability is a rigorous formalism for uncertain knowledge
- Joint probability distribution specifies probability of every atomic event
- Can answer queries by summing over atomic events
- But we must find a way to reduce the joint size for non-trivial domains
- Bayes' rule lets unknown probabilities be computed from known conditional probabilities, usually in the causal direction
- Independence and conditional independence provide the tools

Bayesian Networks

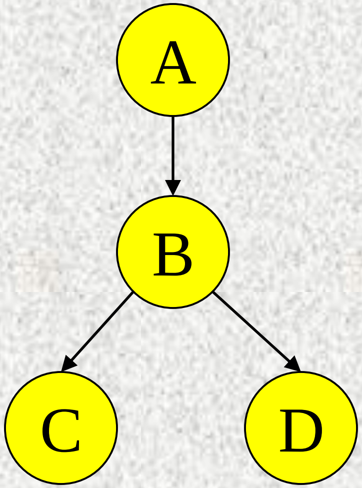


- In the opinion of many AI researchers, Bayesian networks are the most significant contribution in AI in the last 10 years
- They are used in many applications eg. spam filtering, speech recognition, robotics, diagnostic systems and even syndromic surveillance

A Bayesian Network

A Bayesian network is made up of:

1. A Directed Acyclic Graph



2. A set of tables for each node in the graph

A	P(A)
false	0.6
true	0.4

A	B	P(B|A)
false	false	0.01
false	true	0.99
true	false	0.7
true	true	0.3

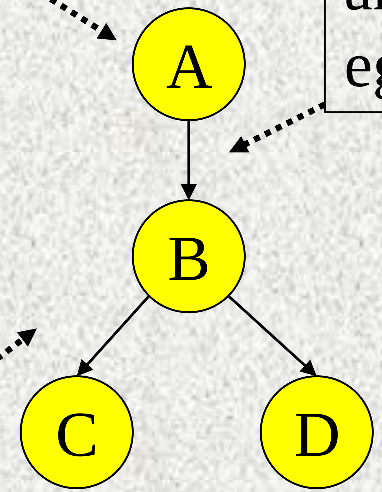
B	D	P(D|B)
false	false	0.02
false	true	0.98
true	false	0.05
true	true	0.95

B	C	P(C|B)
false	false	0.4
false	true	0.6
true	false	0.9
true	true	0.1

A Directed Acyclic Graph

Each node in the graph is a random variable

A node *X* is a parent of another node *Y* if there is an arrow from node *X* to node *Y* eg. *A* is a parent of *B*



Informally, an arrow from node *X* to node *Y* means *X* has a direct influence on *Y*

A Set of Tables for Each Node

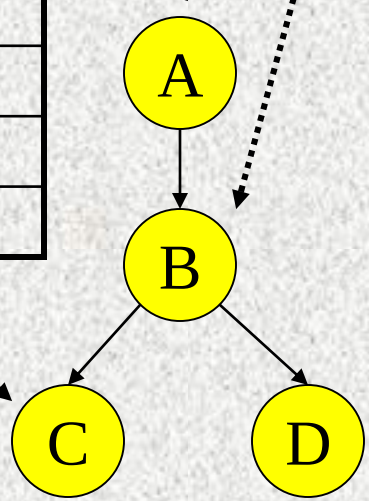
A	P(A)
false	0.6
true	0.4

A	B	P(B|A)
false	false	0.01
false	true	0.99
true	false	0.7
true	true	0.3

Each node X_i has a conditional probability distribution $P(X_i \mid Parents(X_i))$ that quantifies the effect of the parents on the node

The parameters are the probabilities in these conditional probability tables (CPTs)

B	C	P(C|B)
false	false	0.4
false	true	0.6
true	false	0.9
true	true	0.1



B	D	P(D|B)
false	false	0.02
false	true	0.98
true	false	0.05
true	true	0.95

A Set of Tables for Each Node

Conditional Probability Distribution for C given B

B	**C**	**P(C|B)**
false	false	0.4
false	true	0.6
true	false	0.9
true	true	0.1

For a given combination of values of the parents (B in this example), the entries for P(C=true | B) and P(C=false | B) must add up to 1
eg. P(C=true | B=false) + P(C=false |B=false)=1

If you have a Boolean variable with k Boolean parents, this table has 2^{k+1} probabilities (but only 2^k need to be stored)

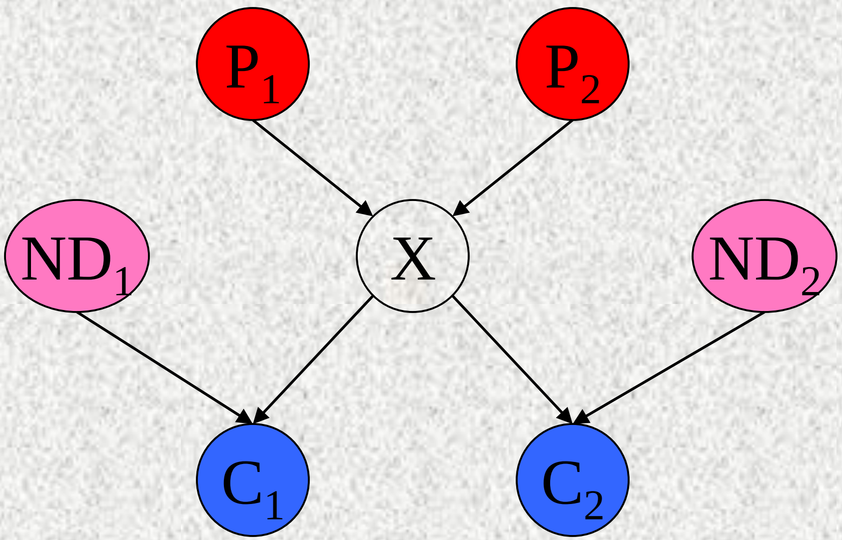
Bayesian Networks

Two important properties:

1. Encodes the conditional independence relationships between the variables in the graph structure
2. Is a compact representation of the joint probability distribution over the variables

Conditional Independence

The Markov condition: given its parents (P_1, P_2), a node (X) is conditionally independent of its non-descendants (ND_1, ND_2)



The Joint Probability Distribution

Due to the Markov condition, we can compute the joint probability distribution over all the variables X_1, …, X_n in the Bayesian net using the formula:

$$P(X_1 = x_1, \ldots, X_n = x_n) = \prod_{i=1}^{n} P(X_i = x_i \mid Parents(X_i))$$

Where Parents(X_i) means the values of the Parents of the node X_i with respect to the graph

Using a Bayesian Network Example

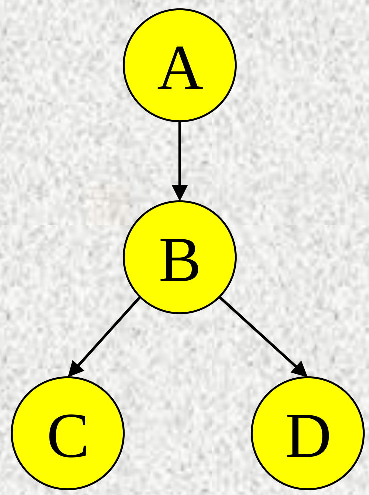
Using the network in the example, suppose you want to calculate:

P(A = true, B = true, C = true, D = true)

= P(A = true) * P(B = true | A = true) *

P(C = true | B = true) P(D = true | B = true)

= (0.4)*(0.3)*(0.1)*(0.95)



Using a Bayesian Network Example

Using the network in the example, suppose you want to calculate:

P(A = true, B = true, C = true, D = true)

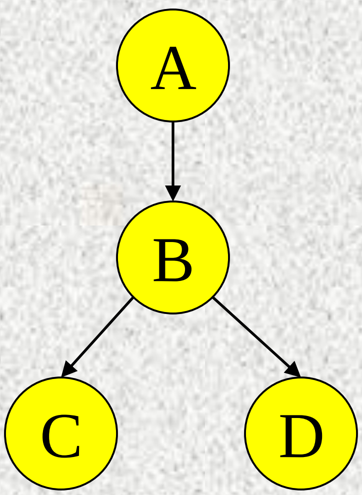
= P(A = true) * P(B = true | A = true) *

P(C = true | B = true) P(D = true | B = true)

= (0.4)*(0.3)*(0.1)*(0.95)

This is from the graph structure

These numbers are from the conditional probability tables



Inference

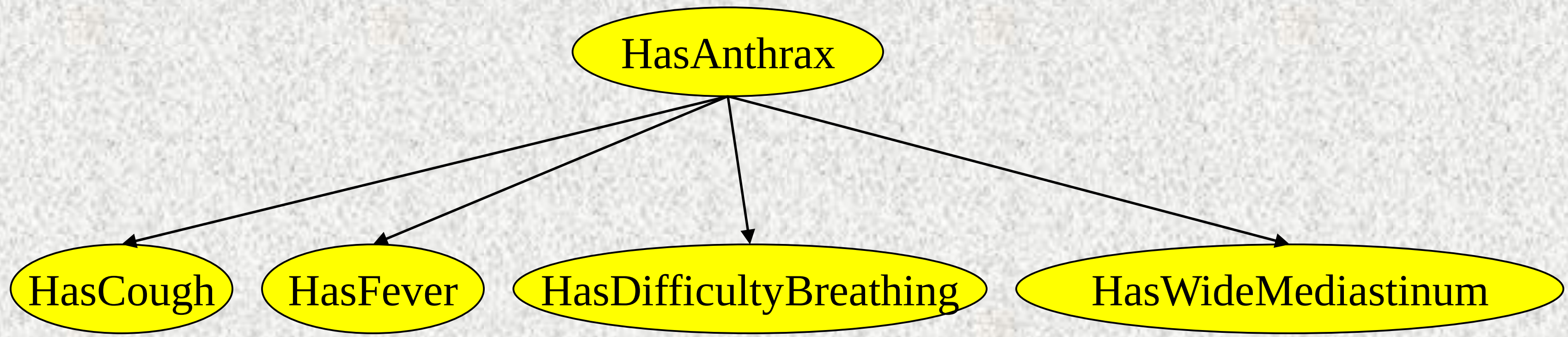
- Using a Bayesian network to compute probabilities is called inference
- In general, inference involves queries of the form:

P(X | E)

E = The evidence variable(s)

X = The query variable(s)

Inference



- An example of a query would be:

  P(*HasAnthrax* = *true* | *HasFever* = *true*, *HasCough* = *true*)
- Note: Even though *HasDifficultyBreathing* and *HasWideMediastinum* are in the Bayesian network, they are not given values in the query (ie. they do not appear either as query variables or evidence variables)
- They are treated as unobserved variables

The Bad News

- Exact inference is feasible in small to medium-sized networks
- Exact inference in large networks takes a very long time
- We resort to approximate inference techniques which are much faster and give pretty good results

Semantic Nets, Frames,

Knowledge Representation as a medium for human expression

- An intelligent system must have KRs that can be interpreted by humans.
  - We need to be able to encode information in the knowledge base without significant effort.
  - We need to be able to understand what the system knows and how it draws its conclusions.

Semantic Networks

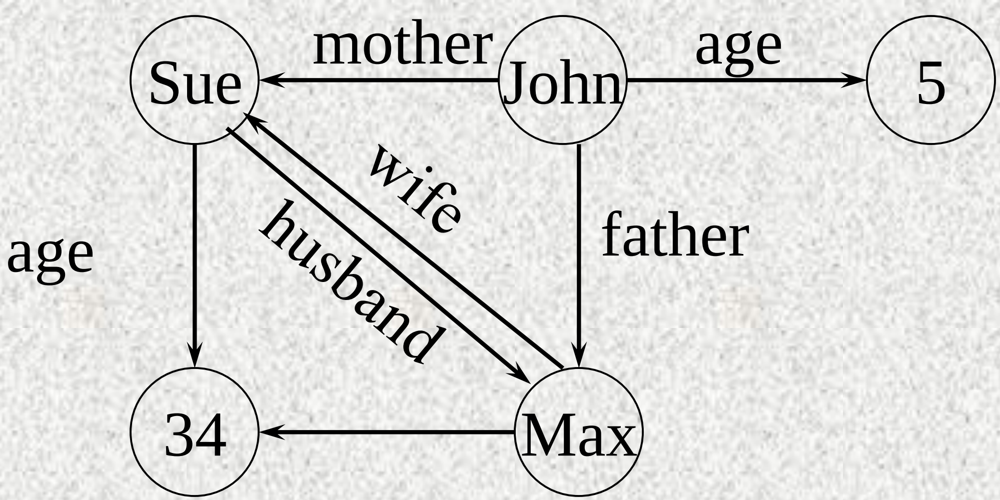
- First introduced by Quillian back in the late-60s

  M. Ross Quillian. "Semantic Memories", In M. M. Minsky, editor, *Semantic Information Processing,* pages 216-270. Cambridge, MA: MIT Press, 1968

- **Semantic network** is simple representation scheme which uses a graph of labeled nodes and labeled directed arcs to encode knowledge
  - Nodes – objects, concepts, events
  - Arcs – relationships between nodes
- **Graphical depiction** associated with semantic networks is a big reason for their popularity

Nodes and Arcs

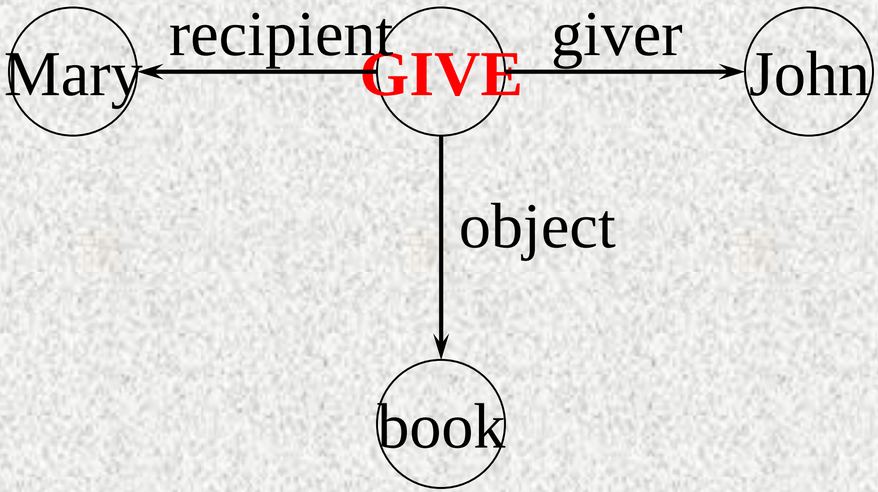
- Arcs define binary relations which hold between objects denoted by the nodes.



mother (john, sue)
age (john, 5)
wife (sue, max)
age (max, 34)
…

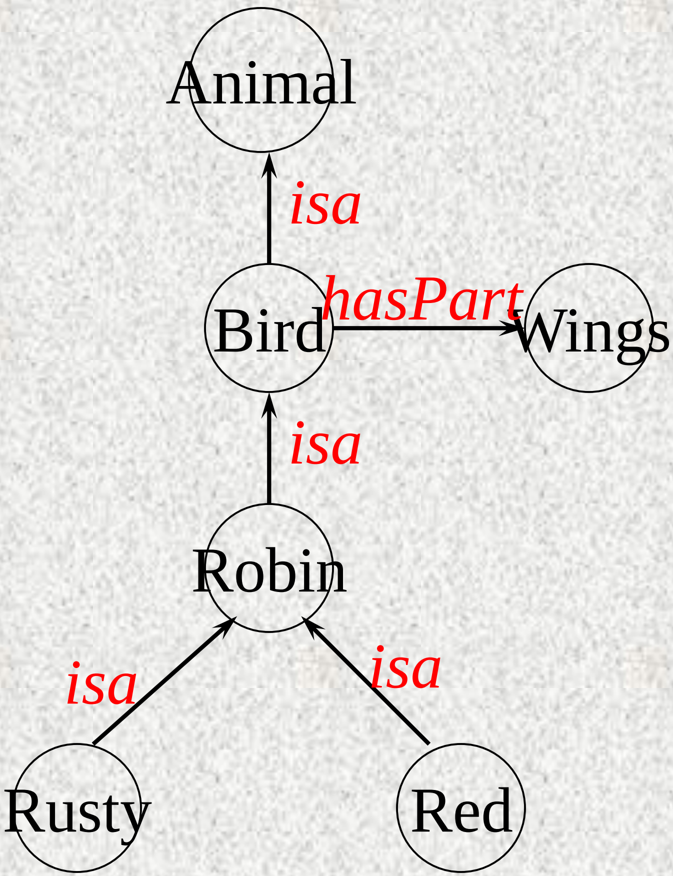
Non-binary relations

- We can represent the generic *give* event as a relation involving three things:
  - A giver
  - A recipient
  - An object



Inheritance

- Inheritance is one of the main kind of reasoning done in semantic nets
- The **ISA** (is a) relation is often used to link a class and its superclass.
- Some links (e.g. **haspart**) are inherited along **ISA** paths
- The semantics of a semantic net can be relatively informal or very formal
  - Often defined at the implementation level

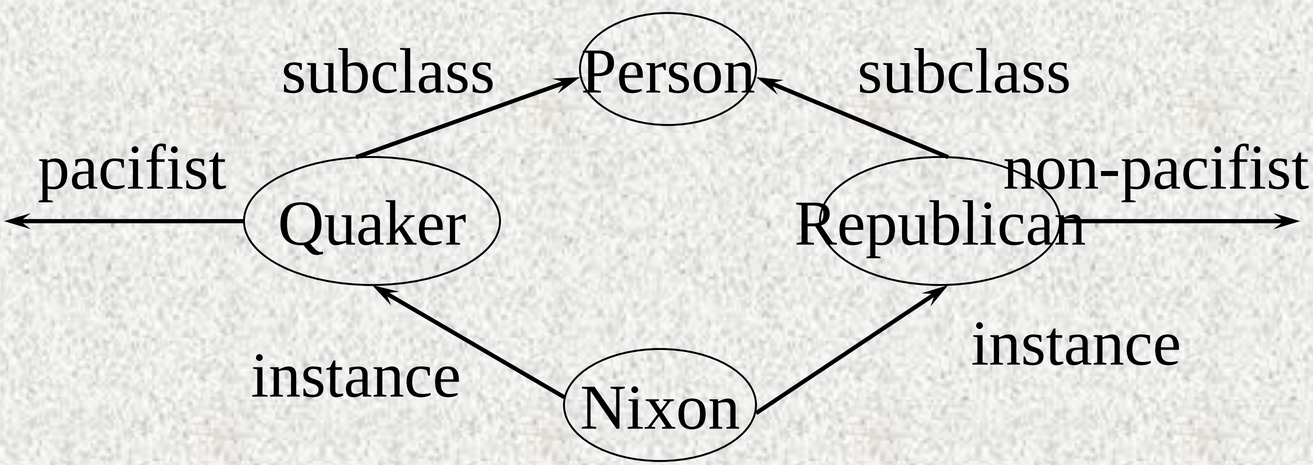


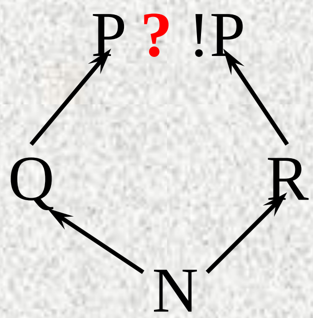
Multiple Inheritance

- A node can have any number of superclasses that contain it, enabling a node to inherit properties from multiple *parent* nodes and their ancestors in the network. It can cause conflicting inheritance.

Nixon Diamond

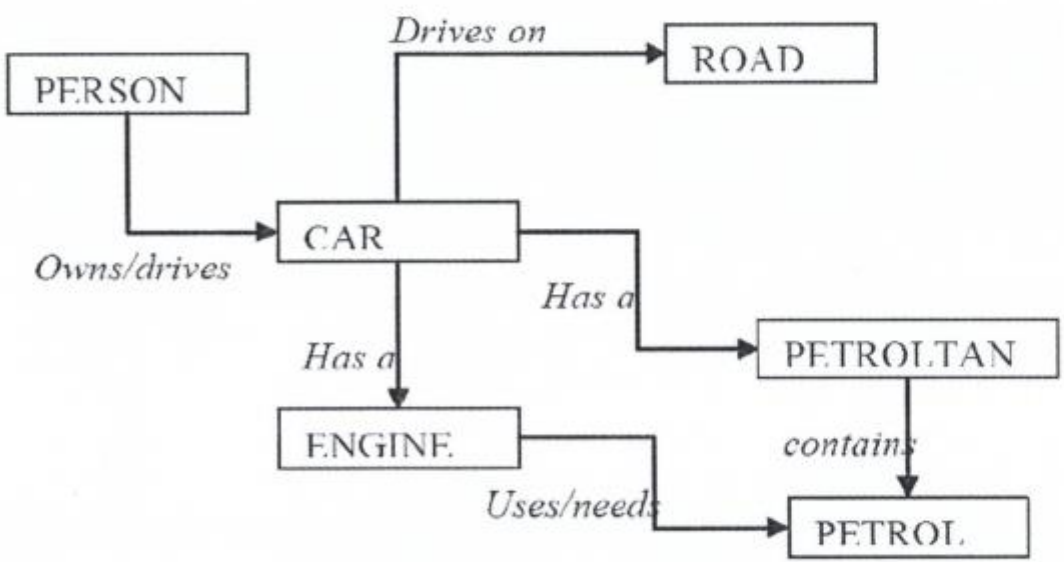
(two contradictory inferences from the same data)





Example

Create a semantic network to describe a car. Your network should include these concepts: *car, person, driver, engine, petrol, petrol tank,* and *road.*



Advantages of Semantic nets

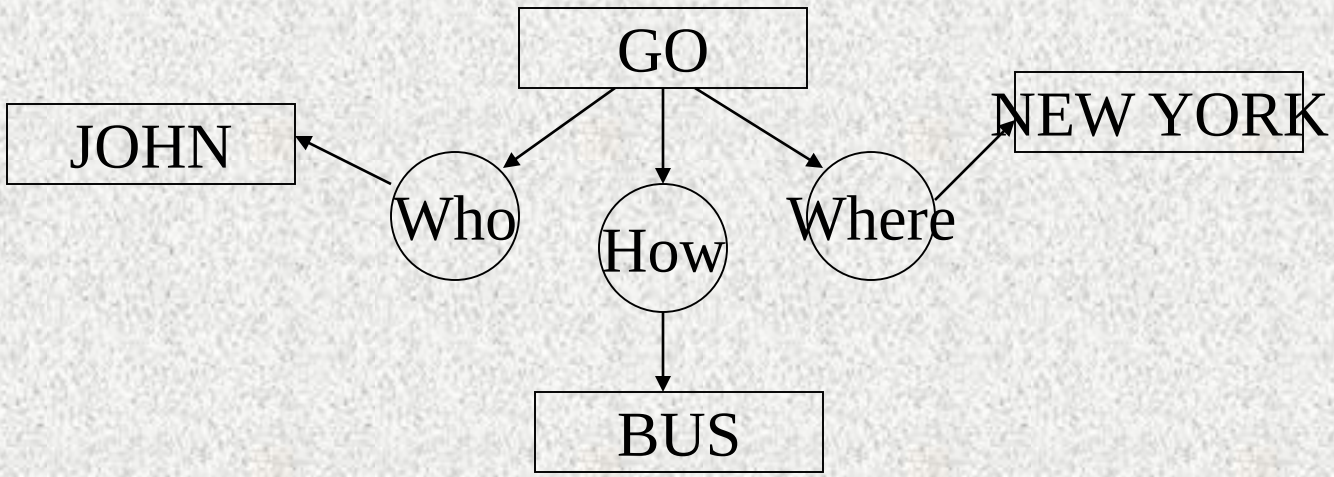
- Easy to visualize
- Formal definitions of semantic networks have been developed.
- Related knowledge is easily clustered.
- Efficient in space requirements
  - Objects represented only once
  - Relationships handled by pointers

Disadvantages of Semantic nets

- Inheritance (particularly from multiple sources and when exceptions in inheritance are wanted) can cause problems.
- Facts placed inappropriately cause problems.
- No standards about node and arc values

Conceptual Graphs

- *Conceptual graphs* are semantic nets representing the meaning of (simple) sentences in natural language
- Two types of nodes:
  - *Concept nodes*; there are two types of concepts, individual concepts and generic concepts
  - *Relation nodes*(binary relations between concepts)



Frames

- Frames – semantic net with properties
- A frame represents an entity as a set of slots (attributes) and associated values
- A frame can represent a specific entry, or a general concept
- Frames are implicitly associated with one another because the value of a slot can be another frame

3 components of a frame

•frame name

•attributes (slots)

•values (fillers: list of values, range, string, etc.)

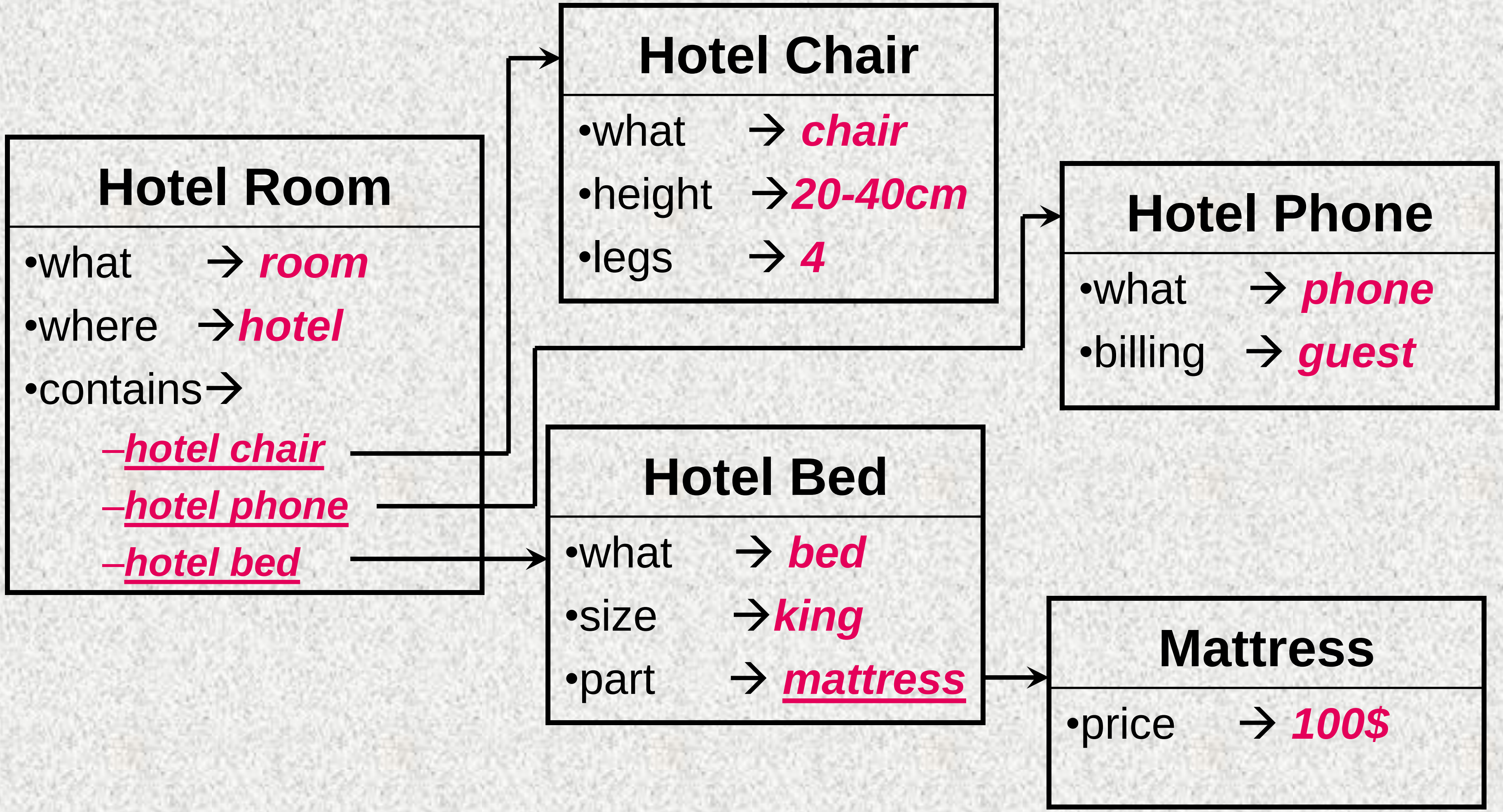
Book Frame
Slot → ***Filler***
•Title → ***AI. A modern Approach*** •Author → ***Russell & Norvig*** •Year → ***2003***

Features of Frame Representation

- More natural support of values then semantic nets (each slots has constraints describing legal values that a slot can take)
- Can be easily implemented using object-oriented programming techniques
- Inheritance is easily controlled

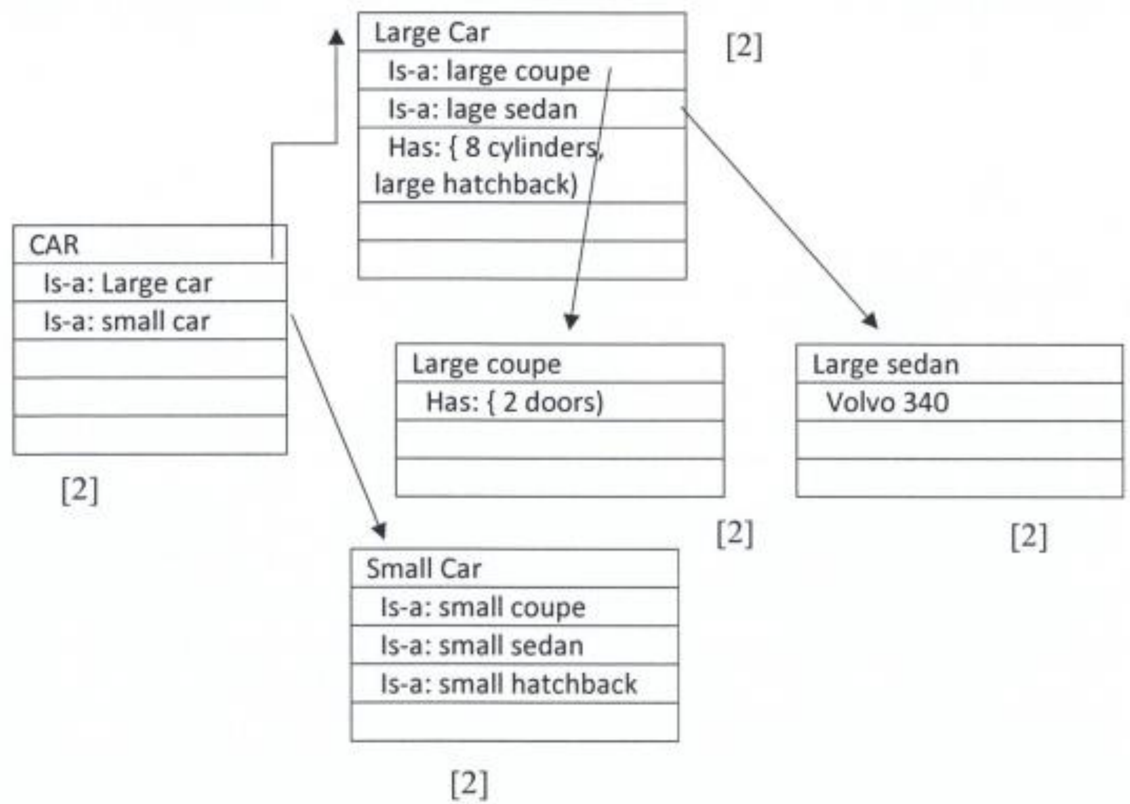
Inheritance

- Similar to Object-Oriented programming paradigm



Examples

"Cars can be classified as small cars and large cars. Small cars include small coupe, small sedan and small hatchback. Large cars are also divided into three types – large coupe, large sedan and large hatchback. Large coupe however has two doors only. Volvo 340 is an example of large sedan cars. All large cars have eight cylinders."



Benefits of Frames

- Makes programming easier by grouping related knowledge
- Easily understood by non-developers
- Expressive power
- Easy to set up slots for new properties and relations
- Easy to include default information and detect missing values

Drawbacks of Frames

- No standards (slot-filler values)
- More of a general methodology than a specific representation:
  - Frame for a class-room will be different for a professor and for a maintenance worker
- No associated reasoning/inference mechanisms

Description Logic

- There is a family of frame-like KR systems with a formal semantics
  - KL-ONE, Classic
- A subset of FOL designed to focus on categories and their definitions in terms of existing relations. **Automatic classification**
  - **Finding the right place in a hierarchy of objects for a new description**
- More expressive than frames and semantic networks
- Major inference tasks:
  - Subsumption

    Is category C1 a subset of C2?
  - Classification

    Does Object O belong to C?

KL-ONE (Brachman, 1977)

•Bi-partite view of knowledge representation

1. Descriptions
2. Assertions

•Entities can be “described” without making any particular assertions about them

•Descriptions are made from other descriptions using a very small set of operators

KL-ONE basics

- Structured inheritance network
- Basic elements:
  - Concepts: Things in the world
    - Generic concepts
    - Individuals
  - Roles: Conceptual properties of an entity
    - parts, attributes, function arguments, linguistic cases
  - Structured descriptions: Relations among roles

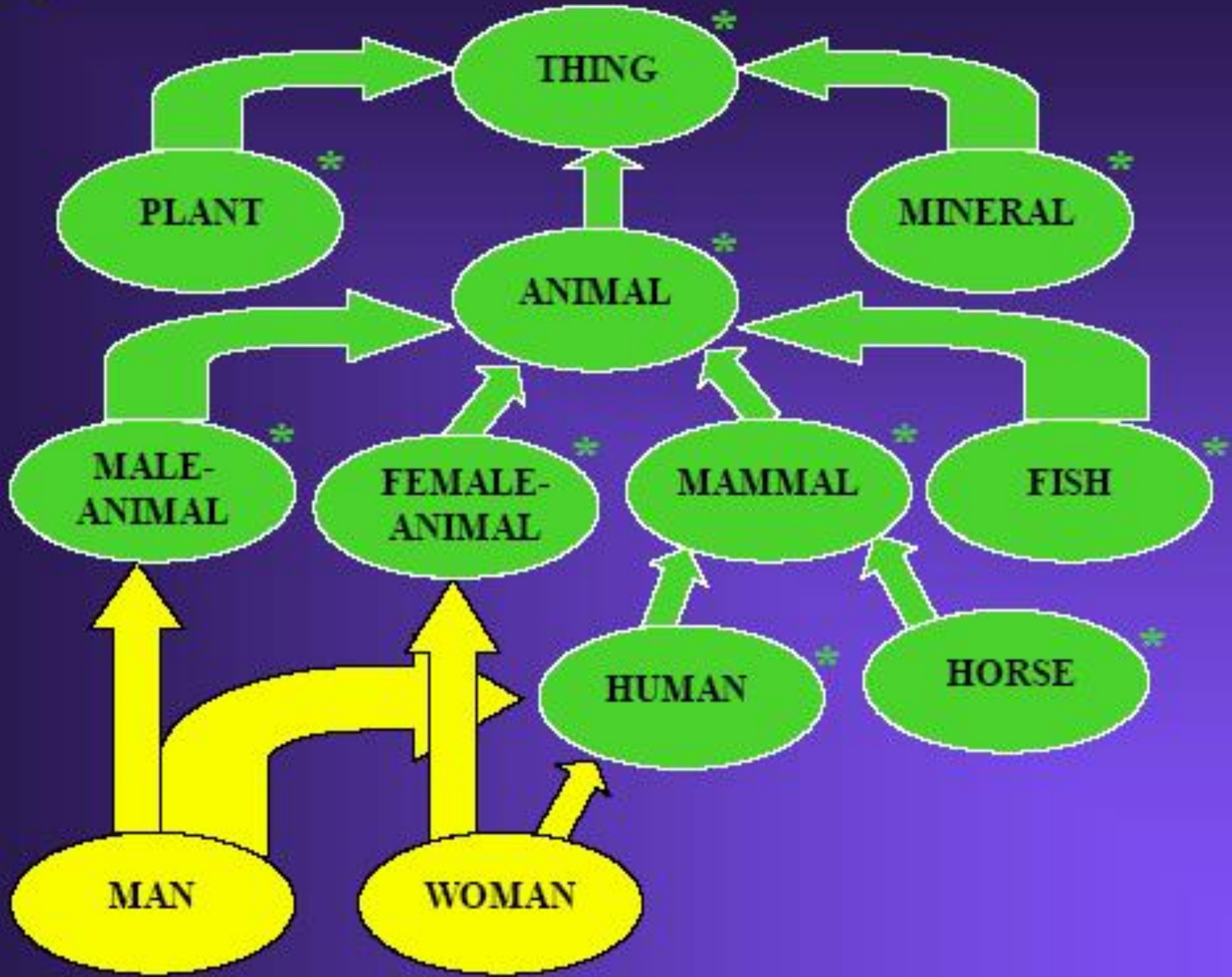
Kinds of concepts

- Defined
  - Have explicit necessary and sufficient properties (roles)
  - Often are specializations of primitive concepts
- Primitive
  - Have no sufficient properties
  - May have other, necessary properties
  - Correspond to *natural kinds*

A KL-ONE Network

- Can be viewed as a kind of semantic network
- Preserves a complex set of relations among descriptions as concepts become more general and more specific
- Clarifies which concepts *subsume* other concepts
- Requires a *classifier* to take new descriptions and to place them where they belong, maintaining all appropriate relationships

A simple KL-ONE network of Generic Concepts

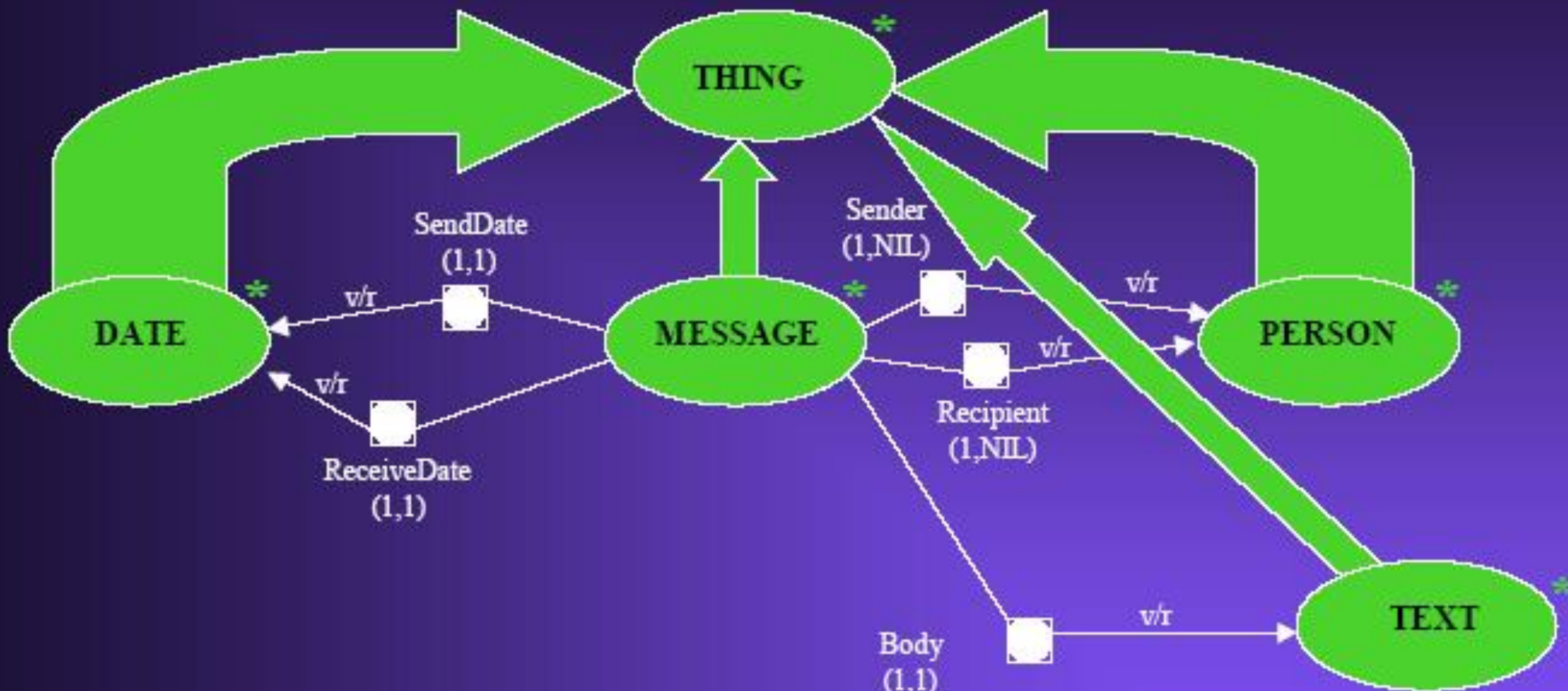


Defined concepts are in yellow;
Primitive concepts are in green.

KL-ONE "Roles"

- Are like *properties* of frames
- Capture the notion that, at different times, a functional role may be played by different entities
- Include value restrictions, which are *necessary* type restrictions on role fillers
- Include number restrictions, which are *necessary* restrictions on cardinality (min, max)

The Primitive Concept MESSAGE



A MESSAGE is, among other things, a THING with at least one Sender, all of which are PERSONs, at least one Recipient, all of which are PERSONs, a Body, which is a TEXT, a SendDate, which is a DATE, and a ReceivedDate, which is a DATE.