Knowledge Representation

Fabio G. Cozman

Escola Politécnica

Universidade de São Paulo

AI's current success:

more power, more data, some insights

AI's current success:
more power, more data, some insights

Knowledge Representation & Reasoning

Decision Making

Machine Learning

Current focus on machine learning

Knowledge Representation & Reasoning

Decision Making

Machine Learning

Statistical

Bio-inspired

Representation learning

What is knowledge?

2019

Knowledge: true belief with explanation.

Platão, Theaetetus

What is knowledge?

2019

Davis et al 1993

Knowledge: true belief with explanation.

Knowledge in AI:
a very flexible thing...
including deterministic facts/rules,
and uncertain beliefs.

Platão, Theaetetus

Knowledge representation

- Often we can solve a problem by coding the algorithm that finds the solution.
  - To do so we must represent the aspects of the problem that are relevant.
- Sometimes it is better to store the relevant facts and rules and use them as needed.
  - The result is a *knowledge-based* system.

Knowledge representation (Davis et al 1993):

- a model ("surrogate"),
- a set of ontological commitments,
- a basis for reasoning,
- a medium for computation.

What Is a Knowledge Representation?

Randall Davis, Howard Shrobe, and Peter Szolovits

■ Although knowledge representation is one of the central and, in some ways, most familiar concepts in AI, the most fundamental question about it—What is it?—has rarely been answered directly. Numerous papers have lobbied for one or another variety of representation, other papers have argued for various properties a representation should have, and still others have focused on properties that are important to the notion of representation in general.

In this article, we go back to basics to address the question directly. We believe that the answer can best be understood in terms of five important and distinctly different roles that a representation plays, each of which places different and, at times, conflicting demands on the properties a representation should have. We argue that keeping in mind all five of these roles provides a usefully broad perspective that sheds light on some long-standing disputes and can invigorate both research and practice in the field.

What is a knowledge representation? We argue that the notion can best be understood in terms of five distinct roles that it plays, each crucial to the task at hand:

First, a knowledge representation is most fundamentally a *surrogate*, a substitute for the thing itself, that is used to enable an entity to determine consequences by thinking rather than acting, that is, by reasoning about the world rather than taking action in it.

Second, it is a set of ontological commitments, that is, an answer to the question, In what terms should I think about the world?

Third, it is a fragmentary theory of intelligent reasoning expressed in terms of three components: (1) the representation's fundamental conception of intelligent reasoning, (2) the set of inferences that the representation sanctions, and (3) the set of inferences that it recommends.

Fourth, it is a medium for pragmatically efficient computation, that is, the computational environment in which thinking is accomplished. One contribution to this pragmatic efficiency is supplied by the guidance that a representation provides for organizing information to facilitate making the recommended inferences.

Fifth, it is a medium of human expression, that is, a language in which we say things about the world.

Understanding the roles and acknowledging their diversity has several useful consequences. First, each role requires something slightly different from a representation; each accordingly leads to an interesting and different set of properties that we want a representation to have.

Second, we believe the roles provide a framework that is useful for characterizing a wide variety of representations. We suggest that the fundamental mind set of a representation can be captured by understanding how it views each of the roles and that doing so reveals essential similarities and differences.

Third, we believe that some previous disagreements about representation are usefully disentangled when all five roles are given appropriate consideration. We demonstrate the clarification by revisiting and dissecting the early arguments concerning frames and logic.

Finally, we believe that viewing representations in this way has consequences for both research and practice. For research, this view provides one direct answer to a question of fundamental significance in the field. It also suggests adopting a broad perspective on

A distinctive feature of AI

- Concern about representation is key to AI.
  - In contrast, decision-making in Economics does not worry about the description of problems.

A distinctive feature of AI

- Concern about representation is key to AI.
  - In contrast, decision-making in Economics does not worry about the description of problems.

- How can...
  - a problem be described concisely and efficiently?
  - we guarantee that all features can be expressed?
  - we quantify the effect of some modeling choices?

Reasoning

- Once we have a representation for our objects of interest, we can reason about them.
- We can decide how to change them, we can extract some understanding from them.
- The operations are often referred to as “inference”.

Common knowledge and common sense

- The shared understanding we have about the "typical" way things are: must be encoded, stored, processed.

Common knowledge and common sense

- The shared understanding we have about the "typical" way things are: must be encoded, stored, processed.

- Consider the Winograd scheme:
  The trophy did not fit into the suitcase because it was too small.
  What was small? Why?

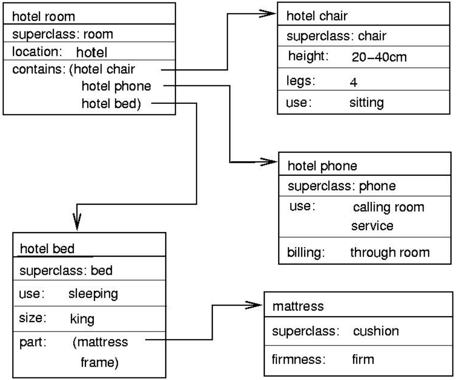
A bit of history: early efforts

- General problem solving by search: problem had to be properly represented.

- *General Problem Solver*: famous effort.
  - Separated declarative knowledge (Horn clausers) from search.
  - A *production system* with if-then rules.

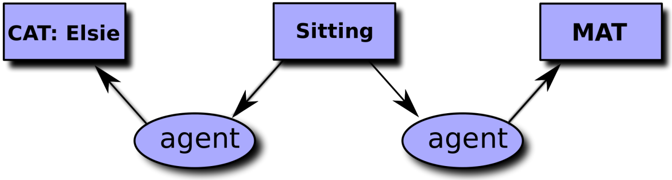
A bit of history: expert systems

- Fever of the eights.
- Idea: capture "expert knowledge" into a knowledge-base.
- Formalisms: production systems, frames.



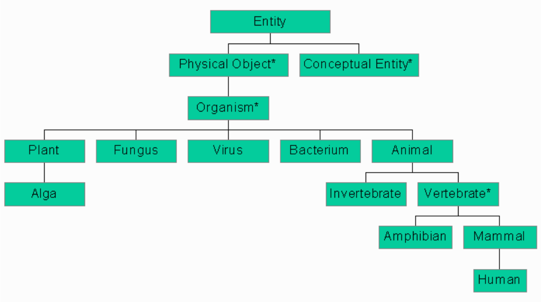
Conceptual graphs

- A graph-based interface to first-order logic.
- A reasoning model based on graphs.



Semantic networks

- Started as graphs in computational linguistics.
- Gradually were formalized using logic.



Expert system shells

- Many systems: PROSPECTOR, CADUCEUS, DENDRAL...
- The famous MYCIN rule-system (shell E-MYCIN):

```
(defrule 52
  if (site culture is blood )
     (gram organism is neg )
     (morphl organism is rod )
     (burn patient is serious)
  then 0.4
     (identity organism is pseudomonas )
)
```

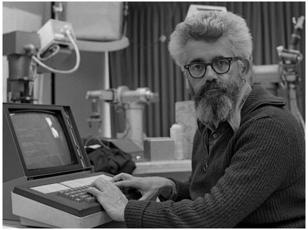
Neats and scruffies

Neats look for elegant solutions, typically with mathematical basis.

Scruffies want to build complex systems that work well.

Victory of the neats

- Gradually, most knowledge representation techniques have found a basis on formal languages.
  - Most techniques are based on logic (propositional, first-order, modal, etc).
- Serious work on complexity and expressivity.

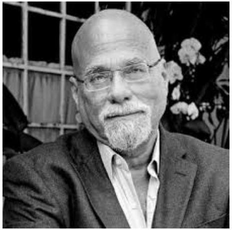


Two successful formalisms

- Description logics.
- Answer set programming.

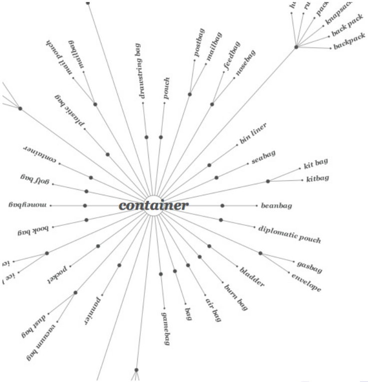
Victory of the scruffies

- Today some of the best tools are very complex and built without major consistency guarantees.
- Examples: WordNet, DBpedia, Freebase, NELL.



WordNet

- Giant multilingual lexical database.
- Free at https://wordnet.princeton.edu/.



DBpedia

- Giant database of facts extracted from Wikipedia.
- Free at https://wiki.dbpedia.org/.
- Data stored in OWL.
- Queries in SPARQL.

OWL in DBpedia

```
<owl:Class rdf:about="http://dbpedia.org/ontology/NationalAnthem">
  <rdfs:label xml:lang="en">National anthem</rdfs:label>
  <rdfs:label xml:lang="fr">Hymne national</rdfs:label>
  <rdfs:label xml:lang="nl">volkslied</rdfs:label>
  <rdfs:comment xml:lang="en">Patriotic musical composition which is the offcial national song.</rd
  <rdfs:subClassOf rdf:resource="http://dbpedia.org/ontology/MusicalWork"/>
  <prov:wasDerivedFrom rdf:resource="http://mappings.dbpedia.org/index.php/OntologyClass:NationalAr
</owl:Class>
```

SPARQL in DBpedia

Example: People who were born in Berlin before 1900.

```
PREFIX dbo: <http://dbpedia.org/ontology/>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
PREFIX : <http://dbpedia.org/resource/>

SELECT ?name ?birth ?death ?person
WHERE { ?person dbo:birthPlace :Berlin .
?person dbo:birthDate ?birth .
?person foaf:name ?name . ?person dbo:deathDate ?death .
FILTER (?birth < "1900-01-01"^^xsd:date) . }
ORDER BY ?name
```

Knowledge graph: NELL

- More than 50M "beliefs" (3M with high confidence).

quaker ave is a highway [confidence 92.8]

Read the Web

Research Project at Carnegie Mellon University

Home | Project Overview | Resources & Data | Publications | People

NELL: Never-Ending Language Learning

Can computers learn to read? We think so. "Read the Web" is a research project that attempts to create a computer system that learns over time to read the web. Since January 2010, our computer system called NELL (Never-Ending Language Learner) has been running continuously, attempting to perform two tasks each day:

- First, it attempts to "read," or extract facts from text found in hundreds of millions of web pages (e.g., `playsInstrument(George_Harrison, guitar)`).
- Second, it attempts to improve its reading competence, so that tomorrow it can extract more facts from the web, more accurately.

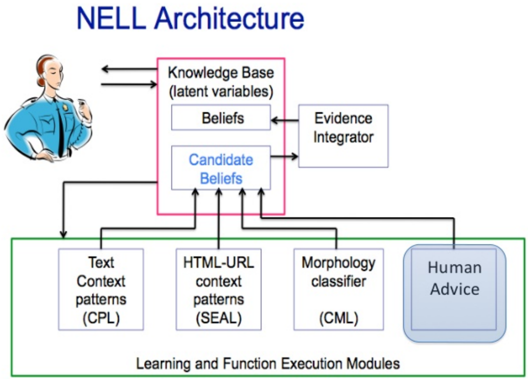
Browse the Knowledge Base!

So far, NELL has accumulated over 50 million candidate beliefs by reading the web, and it is considering these at different levels of confidence. NELL has high confidence in 2,810,379 of these beliefs — these are displayed on this website. It is not perfect, but NELL is learning. You can track NELL's progress below or @cmunell on Twitter, browse and download its knowledge base, read more about our technical approach, or join the discussion group.

Recently-Learned Facts

twitter Refresh

instance	iteration	date learned	confidence
jack_bond is a male	1111	06-jul-2018	90.6
quaker_ave is a highway	1111	06-jul-2018	92.8
ixquick__com is a website	1111	06-jul-2018	92.9
environmental_correspondent is a profession	1111	06-jul-2018	98.2
agco is a biotech company	1111	06-jul-2018	95.7
the sports leage ncaa uses the venue wien_stadium	1111	06-jul-2018	100.0
brewers is a sports team also known as atlanta_braves	1112	24-jul-2018	93.8
rose_garden is a stadium or event venue located in the city portland	1116	12-sep-2018	100.0
national is a company that has an office in the country ireland	1114	25-aug-2018	100.0
peppers is an agricultural product produced in __america	1116	12-sep-2018	99.2



Description logic $\mathcal{ALC}$: basics

- Individuals, concepts, roles.
- *Conjunction* $(C \sqcap D)$, *disjunction* $(C \sqcup D)$, *negation* $(\neg C)$.
- Terminologies (the Tbox): $C \sqsubseteq D$ and $C \equiv D$.

Semantics

- Concept C: set of elements of domain.
- $C \sqcap D$: elements in C and in D.
- $C \sqcup D$: elements in C or in D.
- $\neg C$: elements not in C.

Examples

- $\text{Mother} \sqsubseteq \text{Parent}$.
- $\text{Person} \equiv \text{Human}$.
- $\text{Mother} \equiv \text{Female} \sqcap \text{Parent}$.
- $\text{SchoolPerson} \sqsubseteq \text{Student} \sqcup \text{Professor}$.

$\mathcal{ALC}$ quantification:

- *existential* restriction ($\exists r.C$), and
- *value* restriction ($\forall r.C$).

Semantics

- $\exists r.C = \{x : \exists y : r(x, y) \land C(y)\}$.
- $\forall r.C = \{x : \forall y : r(x, y) \to C(y)\}$.

Examples

- $\text{Concept} \equiv \text{Female} \sqcap \forall\ \text{parentOf.Brazilian}.$
- $\text{Concept} \sqsubseteq \text{Brazilian} \sqcup \exists\ \text{buyFrom.Brazilian}.$

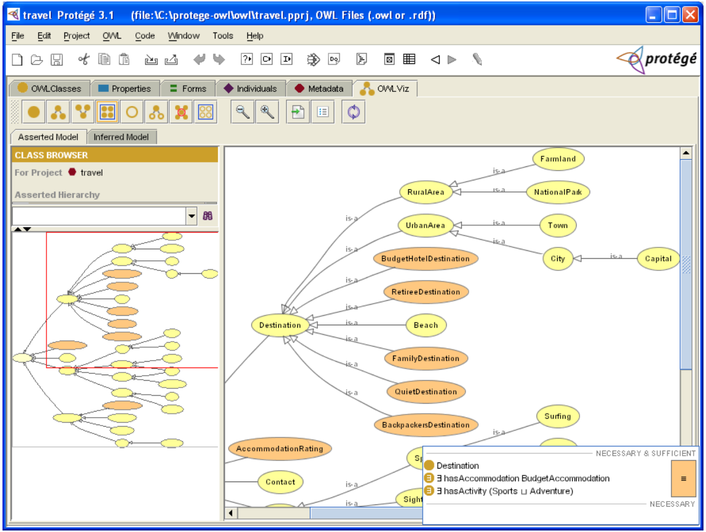
Assertions: the Abox

- Fruit(appleFromJohn),
- buyFrom(Bob, John).

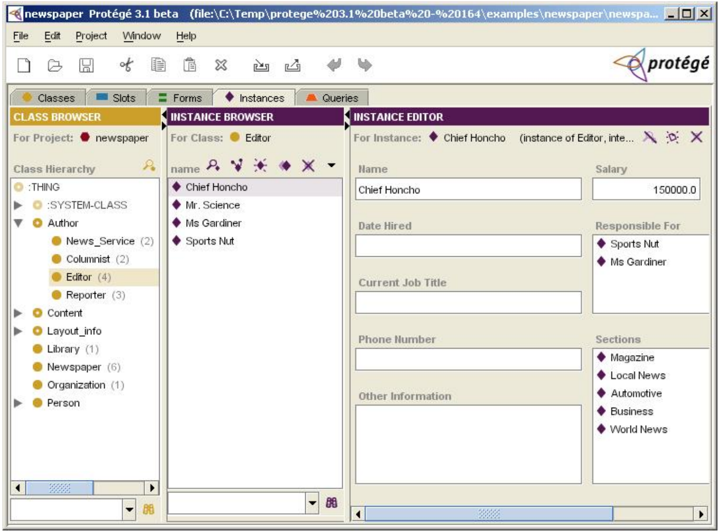
Ontologies

- A set of "axioms" is an ontology (terminology).
- The TBox stores the axioms.
- Assertions are stored in ABox.

Protege system (TBox)



Protege system (ABox)



OWL

- The language OWL, based on description logics, is now the standard for data storage in the *Semantic Web*.
- OWL is based on fragments of $\mathcal{ALC}$; in particular DL-Lite and $\mathcal{EL}$.

Logic programming

- Venerable Prolog language.
- Rules:

$$\text{pass}(X, Y) :- \text{student}(X), \text{adept}(X), \text{course}(Y), \text{easy}(Y).$$

Propositional Horn clauses

- A Horn clause is a rule without negation and a single atom in the head.

$$A :- B_1, B_2, \ldots, B_n.$$

Propositional Horn clauses

- A Horn clause is a rule without negation and a single atom in the head.

$$A :- B_1, B_2, \ldots, B_n.$$

- Read as:

$$B_1 \wedge B_2 \wedge \cdots \wedge B_n \rightarrow A.$$

- Then transform to clause:

$$\neg B_1 \vee \neg B_2 \vee \cdots \vee \neg B_n \vee A.$$

Stratified programs: paths

$$\text{edge}(X, Y) :- \text{edge}(Y, X).$$
$$\text{path}(X, Y) :- \text{edge}(X, Y).$$
$$\text{path}(X, Y) :- \text{edge}(X, Z), \text{path}(Z, Y).$$

Stratified programs: paths

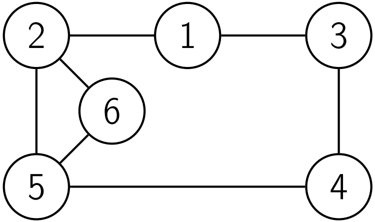
$$\text{edge}(X, Y) \text{ :- } \text{edge}(Y, X).$$
$$\text{path}(X, Y) \text{ :- } \text{edge}(X, Y).$$
$$\text{path}(X, Y) \text{ :- } \text{edge}(X, Z), \text{path}(Z, Y).$$

$$\text{edge}(1, 2).$$
$$\text{edge}(1, 3).$$
$$\text{edge}(2, 5).$$
$$\text{edge}(2, 6).$$
$$\text{edge}(3, 4).$$
$$\text{edge}(4, 5).$$
$$\text{edge}(5, 6).$$

Stratified programs: paths

$$\text{edge}(X, Y) :- \text{edge}(Y, X).$$
$$\text{path}(X, Y) :- \text{edge}(X, Y).$$
$$\text{path}(X, Y) :- \text{edge}(X, Z), \text{path}(Z, Y).$$

edge(1, 2).
edge(1, 3).
edge(2, 5).
edge(2, 6).
edge(3, 4).
edge(4, 5).
edge(5, 6).



Answer Set Programming

$$\text{red}(X) \lor \text{green}(X) \lor \text{blue}(X) :- \text{node}(X).$$
$$\text{edge}(X, Y) :- \text{edge}(Y, X).$$
$$:- \text{edge}(X, Y), \text{red}(X), \text{red}(Y).$$
$$:- \text{edge}(X, Y), \text{green}(X), \text{green}(Y).$$
$$:- \text{edge}(X, Y), \text{blue}(X), \text{blue}(Y).$$

red(1)., green(4)., green(6)..

Three additional issues

- Defaults...
- Non-monotonic reasoning...
- Revisions...

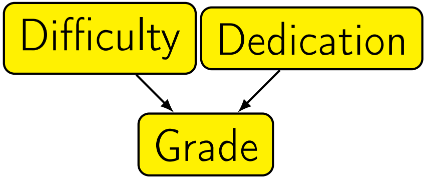
... and another topic: KR in planning...

Representing uncertainty

- At first, probabilities were considered inadequate for AI.
- Situation changed with
  - the appearance of graph-theoretical models (exploit independence to produce compact representations);
  - the use of decision processes in planning under uncertainty;
  - the availability of data and the success of statistical and neural methods.
- Today, probabilities pervade many areas of AI. Other formalisms: belief functions, fuzzy logic.

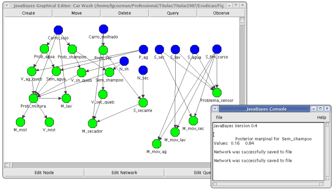
Bayesian networks

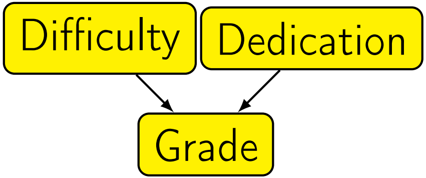
- The language of Bayesian networks (at first KR, then ML...)



Bayesian networks

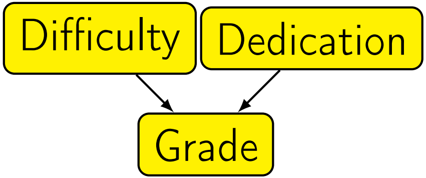
- The language of Bayesian networks (at first KR, then ML...)

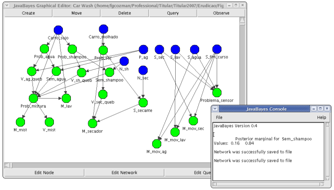


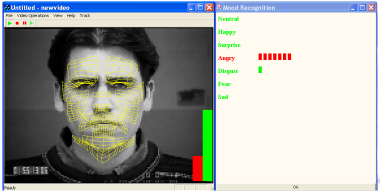


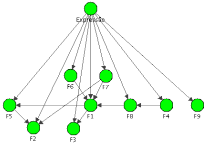
Bayesian networks

- The language of Bayesian networks (at first KR, then ML...)

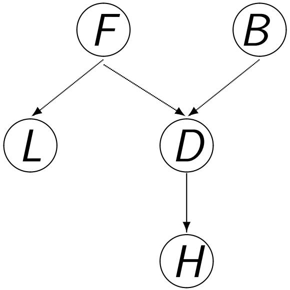








The dog problem (Charniak 1991)



$$\mathbb{P}(f) = 0.5 \qquad \mathbb{P}(b) = 0.5$$
$$\mathbb{P}(l|f) = 0.6 \qquad \mathbb{P}(l|f^c) = 0.05$$
$$\mathbb{P}(d|f,b) = 0.8 \qquad \mathbb{P}(d|f,b^c) = 0.1$$
$$\mathbb{P}(d|f^c,b) = 0.1 \qquad \mathbb{P}(d|f^c,b^c) = 0.7$$
$$\mathbb{P}(h|d) = 0.6 \qquad \mathbb{P}(h|d^c) = 0.3$$

Semantics

- Markov condition: Every variable is independent of its nondescendants nonparents given its parents.
- The Markov condition implies that any Bayesian network represents a unique joint probability density that factorizes as:

$$\mathbb{P}(X_1 = x_1, \ldots, X_n = x_n) = \prod_i \mathbb{P}(X_i = x_i | \mathrm{pa}(X_i) = \pi_i) .$$

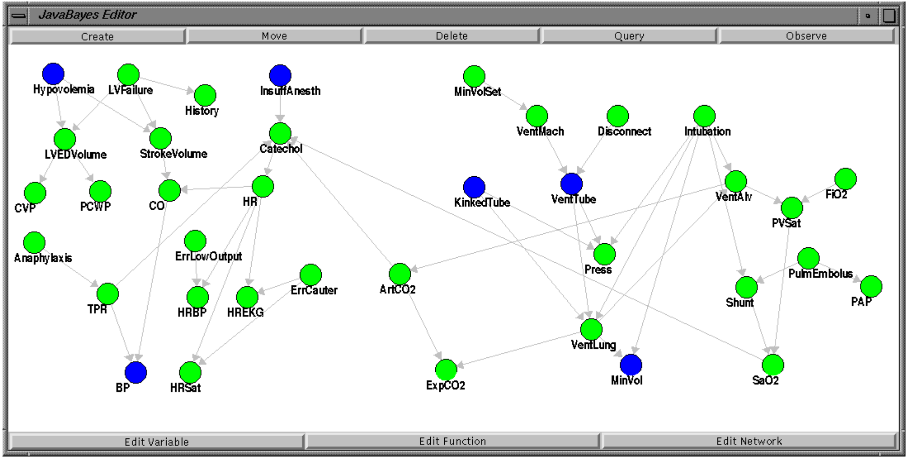
Probabilistic Expert Systems

- Expert Systems tried to address the fact that human understanding depends on stored facts.
- Difficulties in handling uncertainty: the famous MYCIN system, based on rules and *certainty factors*.

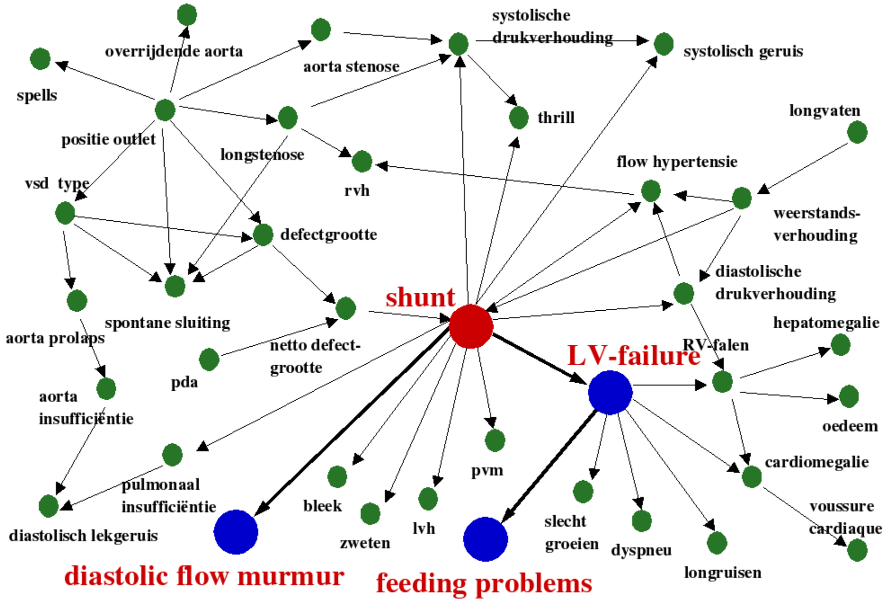
The debate around certainty factors

- Certainty factors were once popular due to the influence of the MYCIN project.
- Interpretation based on Dempster-Shafer theory was once very popular.
- Possible to give probabilistic interpretation (but this was not the point).
- In the end, all of this was very ad hoc.
- Bayesian networks appeared as a reaction to this situation.

The Alarm network



The LV-failure network

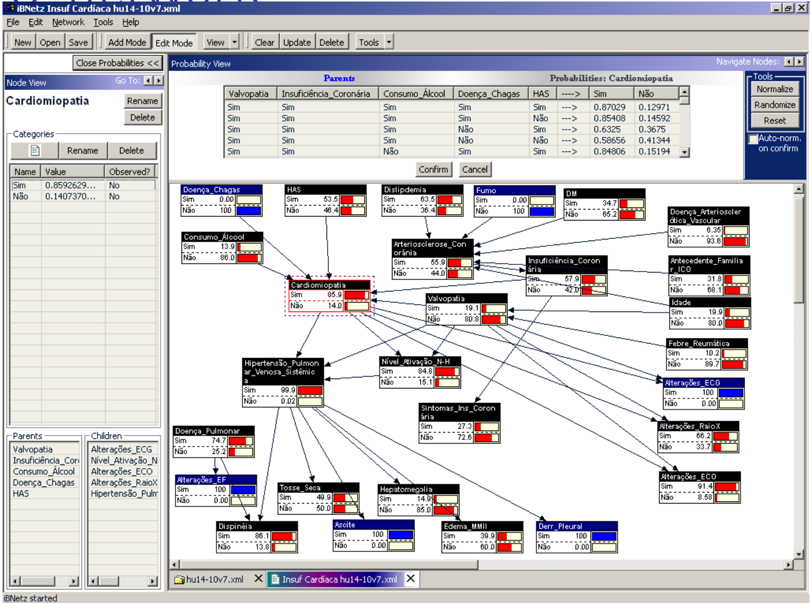


Others...

- Pathfinder
- MUCIN (and then Hugin)
- Hailfinder

Related systems in Economics, Management, Operations Research, ...

The HU network



Elicitation

- Start identifying variables.
  - Discussion: better to use 4 binary variables or a variable with 16 categories?
  - Discussion: better to group 3 binary variables with only 3 possible joint categories into a single variable?
  - Discussion: possible to discretize a variable into a coarse set of categories?
- Then build the graph.
  - Use causality.
  - Avoid cycles! Example: sedentary life causes weak heart causes heart disease causes sedentary life.
- Then elicit the numbers.

Getting the numbers

1. From the literature.
2. From an expert (or panel of experts).

This often takes long; there are several support tools in the literature.

Eliciting numbers

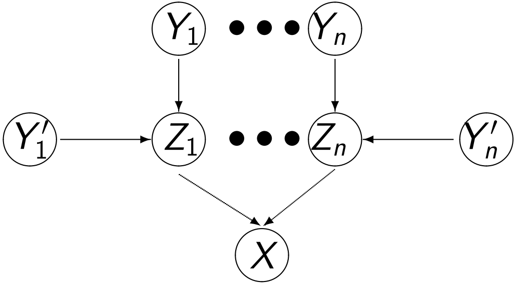
Ideas:

1. Draw a scale: linear or pie chart.
2. Use lotteries: would you rate the chance that this patient gets lung infection as the same that he wins this (specified separately) lottery?
3. Use words, then translate to numbers:
   never - rare - improbable - about half - more than half - often - commonplace - almost always - always.

Modeling tools: Noisy OR

1. If node has many parents, too many parameters to specify.
2. But relationship may be simple (few actual parameters).
3. Most famous case is Noisy OR.
4. Basic idea: "any member of a set of conditions is likely to cause a certain event and this likelihood does not diminish when several of these conditions prevail simultaneously."

The structure of Noisy OR



$$Z_i = Y_i \text{ AND } Y'_i$$

$$X = Z_1 \text{ OR } Z_2 \ldots \text{ OR } Z_n$$

Noisy OR

1. p_k: probability that the kth inhibitor is FALSE:

   $$\mathbb{P}(X = \text{TRUE} | \text{all FALSE but } Y_k = \text{TRUE}) = 1 - p_k.$$

   The p_k are called *link probabilities*.
2. Denote by T the subset of indexes of parents of X that are TRUE in a given configuration; then:

   $$\mathbb{P}(X = \text{TRUE} | Y_1, \dots, Y_n) = 1 - \prod_{i \in T} p_i.$$

Advantages of Noisy OR

- Scheme leads to great simplifications!
  - Less numbers to elicit.
  - The OR node can be decomposed into several OR nodes with two inputs each (simplifying inference).
- Variant: leak probability that X is TRUE even when all Y_k are FALSE.
- There are many generalizations and variants (max, sum, etc).