Knowledge Representation and Reasoning

- We discussed the need to, and a few ways to represent knowledge. Today we'll continue with this and provide examples for how to **reason** with this knowledge.
- Specifically, we talked about various forms of reasoning:

Deduction: Conclusion from given axioms (facts or observations)

All humans are mortal. *Socrates is a human.*	(axiom) (fact/ premise)
Therefore, it follows that Socrates is mortal.	(conclusion)

Induction: Generalization from background knowledge or observations

Socrates is a human *Socrates is mortal*	(background knowledge) (observation/ example)
Therefore, I hypothesize that all humans are mortal	(generalization)

Abduction: Simple and mostly likely explanation, given observations

All humans are mortal *Socrates is mortal*	(theory) (observation)
Therefore, Socrates must have been a human	(diagnosis)

- First, each of these needs to be formalized, so that a computational theory can be developed; and it needs to be studied in the context of various knowledge representations.

- Of these, Abduction might be the one that is most useful (??) and hardest to formalize. (more later).

- But, are these forms of reasoning sufficient?

- People talk about many types of reasoning:
    - Quantitative Reasoning

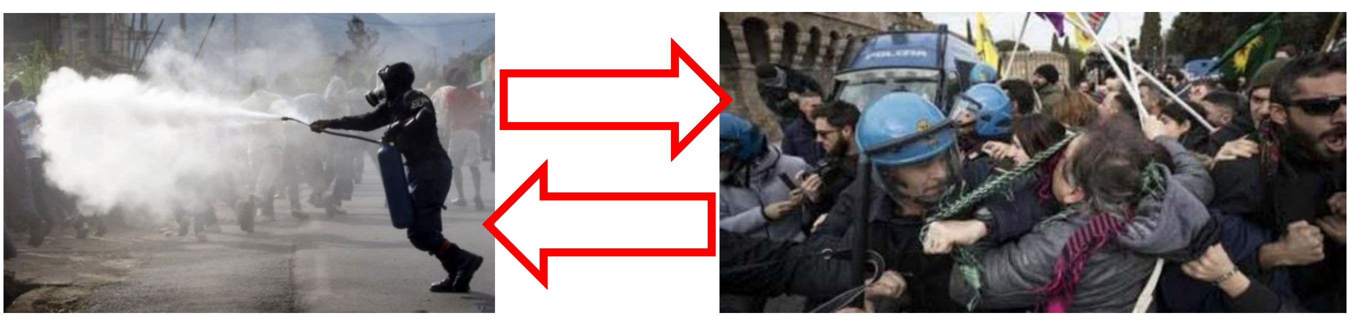
Example: The sum of two numbers is 111. One of the numbers is consecutive to the other number. Find the two numbers.

Example: Bill s father s uncle is twice as old as bills father. 2 years from now bill s father will be 3 times as old as bill. The sum of their ages is 92. Find Bill s age.

Example: The distance between New York to Los Angeles is 3000 miles. If the average speed of a jet place is 600 miles per hour find the time it takes to travel from New York to Los Angeles by jet.

Example: Ram Emanuels' campaign contributions total that of all his competitors together.

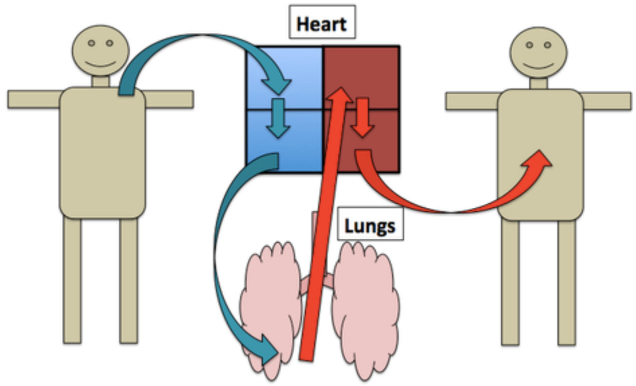
    - Temporal Reasoning
        - I woke up at 8am; I have a meeting in an hour. (When is the meeting?)
        - Duration; order of events
        - "Please get me a piece of cake"



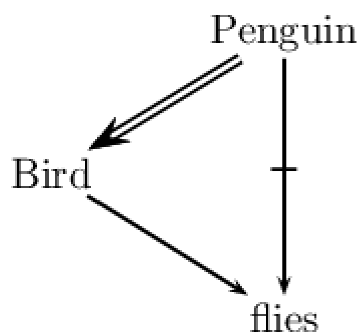
Police used tear gas.

People were angry

    - Causal (cause to effect; effect to cause)

- Analogy
  - The heart is a pump
  - 

-
- Non-monotonic
  - Birds fly
  - Tweedy is a bird; does Tweedy fly?
  - Tweedy is a penguin
  -



- Are these all different phenomena? Do they require different formalisms?
- (and we are not talking yet about knowledge acquisition; we will do this as we introduce papers in this area).

Flows of ideas:

- Progress occurred in multiple research communities:
  - Mostly in KR&R, where the effort was to develop general paradigms, under the assumption that NLP is just an application.
  - In NLP (and in other applications, such as Planning, Robotics, some in Vision)

Logic: $\mathbf{KB} \models \alpha$

First order logic → (too complex to compute) Propositional logic

- **Idea:** represent all your knowledge in FOL (**KB**).
- **Given a query α, determine whether it holds in the KB: (KB implies α)**

Facts:	**(Declarative) Knowledge:**
• Joe is married to Sue • Bill has a brother with no children. • Henry's friends are Bill's cousins.	• *Ancestor* is the transitive closure of *parent*. • *Brother* is *sibling* restricted to males • *Favourite-cousin* is a special type of *cousin*.

Representation:

$$\forall x\, \mathsf{Friend}(\mathsf{henry}, x) \equiv \mathsf{Cousin}(\mathsf{bill}, x)$$

- **Problem I: complexity of inference.**
-
- **(but of, course, there were many other problems** – incomplete knowledge, expressiveness, uncertainty)

This gave rise to a large number of representations

- Limited forms of FOL.
- Relations Databases

Course(csc248)	Dept(csc248,ComputerScience)	Enrollment(csc248,42)
Course(mat100)	Dept(mat100,Mathematics)	

- Where the hope what the you will be able to address questions such as:

How many courses are offered by the Computer Science Department?

- But there were many other representations languages that were developed, some along with inference systems.

- **Logic program (Prolog): a collection of Horn sentences**

$$\forall x_1 \cdots x_n [P_1 \wedge \cdots \wedge P_m \supset P_{m+1}] \quad \text{where} \quad m \geq 0 \text{ and each } P_i \text{ is atomic}$$

For example:

```
parent(bill,mary).
parent(bill,sam).
mother(X,Y) :- parent(X,Y), female(Y)
female(mary).
```

Now I know who is the **mother of Bill** (if I execute the program)

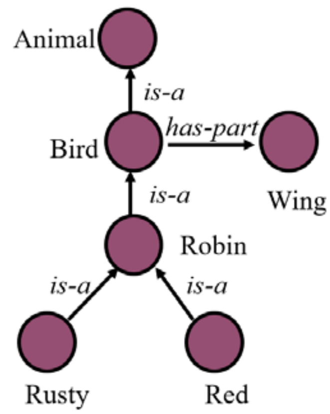
- **This direction addresses expressivity, and traded it of with tractability**

    - Propositional Logic (Boolean formulas oven a set of Boolean variables)
        - Horn logic

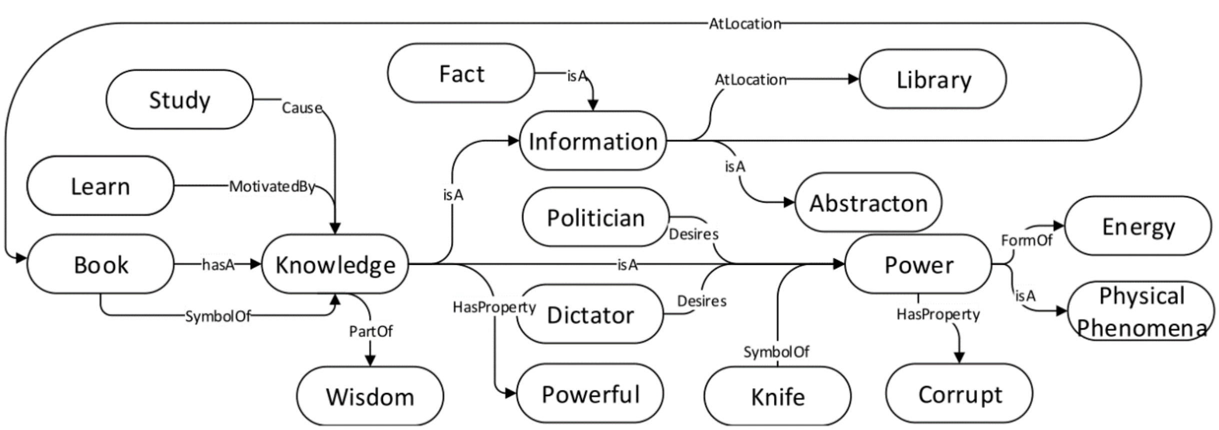
Problem II: Expressivity

- Semantic Networks: allows the use of more expressive predicates, and more “intuitive inference”. People talked about inference as a form of “spreading activation”
    - A graph of labeled nodes and labeled, directed arcs
    - Arcs define relationships that hold between objects denoted by the nodes.
    - Nodes can have multiple attributes.

Link Type	Semantic s	Example
$A \xrightarrow{Subset} B$	$A \subset B$	$Cats \subset Mammals$
$A \xrightarrow{Member} B$	$A \in B$	$Bill \in Cats$
$A \xrightarrow{R} B$	$R(A,B)$	$Bill \xrightarrow{Age} 12$
$A \xrightarrow{\boxed{R}} B$	$\forall x, x \in A \Rightarrow R(x,B)$	$Bird \xrightarrow{\boxed{legs}} 12$
$A \xrightarrow{\boxed{\boxed{R}}} B$	$\forall x\, \exists y, x \in A \Rightarrow y \in B \wedge R(x,B)$	$Birds \xrightarrow{\boxed{\boxed{Parent}}} Birds$



- This went in two directions:
- **Concept nets:**
    - Based on Open Mind Common Sense (OMCS)
    - Intended to serve as a large commonsense knowledge base
    - Built on contributions of many people across the Web.



- More importantly, formalized in terms of **Description Logics**, and then elaborated into **Frame Description Forms**.

- **Frames** were used to describe types and their attributes: values, Restrictions, attached procedures (how an attribute should be used).

  (Student
  with a dept **is** computer-science **and**
  with $\geq$ 3 enrolled-course **is a**
  (Graduate-Course
  with a dept **is a** Engineering-Department))

  - Eventually, this led to theories of **Frames** (Minsky), and **Scripts** (Schank)

- More generally, these languages had expressive grammars:

$$
\begin{aligned}
\langle type\rangle ::= &\ \langle atom\rangle \\
&\mid (\texttt{AND}\ \langle type_1\rangle \ldots \langle type_n\rangle) \\
&\mid (\texttt{ALL}\ \langle attribute\rangle\ \langle type\rangle) \\
&\mid (\texttt{SOME}\ \langle attribute\rangle)
\end{aligned}
$$

$$
\begin{aligned}
\langle attribute\rangle ::= &\ \langle atom\rangle \\
&\mid (\texttt{RESTR}\ \langle attribute\rangle\ \langle type\rangle)
\end{aligned}
$$

- Example: The set of all people the all their male friends are doctors with some specialty.

  (AND person (ALL (RESTR friend male) (AND doctor (SOME specialty)))).

- Came with inference algorithms – **subsumption,** and was extremely influential – all systems built in the 80-ith and later, were built on these languages. It was also influential in areas such as Feature Extraction for machine learning, and theories of grammar.

Problem II: Expressivity

What about Probabilities?

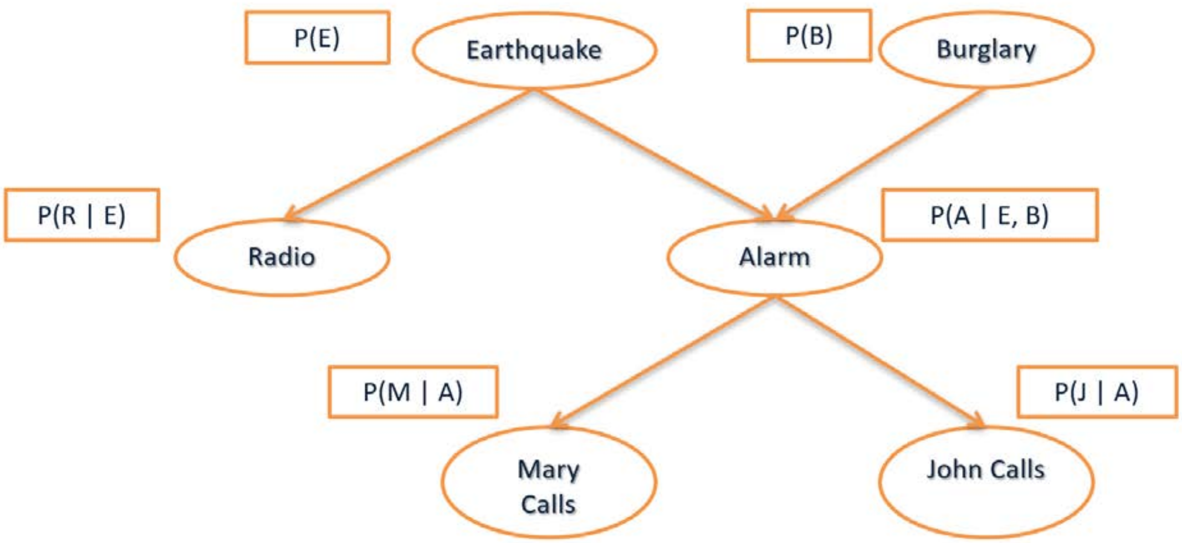
- In parallel to the progress on the logical representations, people argued that we need to deal with uncertainly, and need to move to **probabilistic representations.**
- Progress here proceeded in two camps
  - (Propositional) representation of distributions
    - **Bayesian Networks** (Pearl 1988)
  - Probabilistic extensions of the FOL/Prolog representations. (Haddawy 1993)
    - **Problog**
    - Markov Logic Network

- Two important comments:
  - The latter direction is presented today as fusing probabilities with declarative (logical) knowledge. This, in fact, was studies much earlier (in the 60—ies), but without practical implementations.
  - Fusing Probabilities with Declarative information is different from fusing Learning with Declarative Information. In fact, none of the bullets above came with a native approach for **learning**.
  - Fusing learning with declarative knowledge came later in the context of Structured Learning, e.g., ILP Formulations, Roth & Yih 2004, and following work.

Probabilistic Representations:

Bayes Nets:

- Nodes are random variables
- Edges represent causal influences
- Each node is associated with a conditional probability distribution



- **Computational Problems (**Inference):
  - **Computing the probability of an event:**
  - Given structure and parameters
  - Given an observation E, what is the probability of assignment Y?
  - **P(R=off, A=off | E=e) =?** (E, Y are sets of instantiated variables)
- **Most likely explanation (Maximum A Posteriori assignment, MAP, MPE)**
  - Given structure and parameters
  - Given an observation E, what is the most likely assignment to Y?
  - Argmax_y **P(Y=y | E=e) (Say, Y = (R, A))**
  - (E, Y are sets of instantiated variables)

Probabilistic Relational Representations:

- Representation of distribution over relations, as opposed to propositional variables.
- Ability to build programs that do not only encode **complex interactions** between variables, but also expresses the inherent **uncertainties**.

```
0.3::stress(X) :- person(X).
0.2::influences(X,Y) :- person(X), person(Y).

smokes(X) :- stress(X).
smokes(X) :- friend(X,Y), influences(Y,X), smokes(Y).

0.4::asthma(X) :- smokes(X).

person(angelika).
person(joris).
person(jonas).
person(dimitar).

friend(joris,jonas).
friend(joris,angelika).
friend(joris,dimitar).
friend(angelika,jonas).
```

Inference: Becoming much harder. For the most part, done by **propositionalizing** relational representations (that is, substitution of all domain variables, and blowing up the representations to get a propositional BN). But, there are other ways, e.g., lifted inference.

Next: **Learning with Declarative Representations.**