CSC421 Intro to Artificial Intelligence

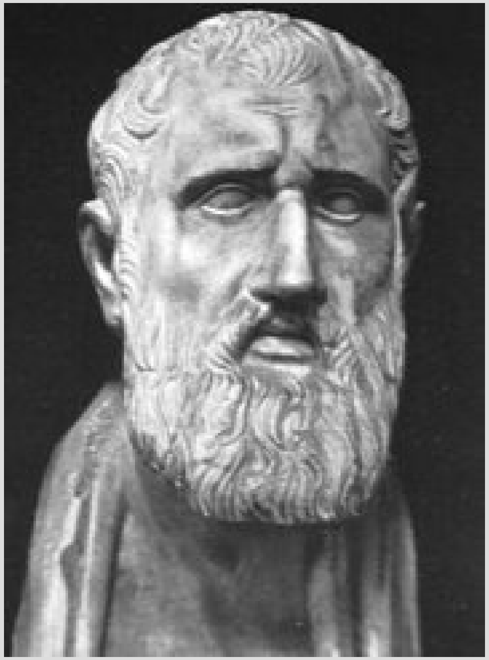
UNIT 13:
Inference in First-Order Logic

Outline

- Reducing first-order inference to propositional inference
- Unification
- Generalized Modus Ponens
- Forward and backward chaining
- Resolution
- Logic Programming

Brief history of reasoning

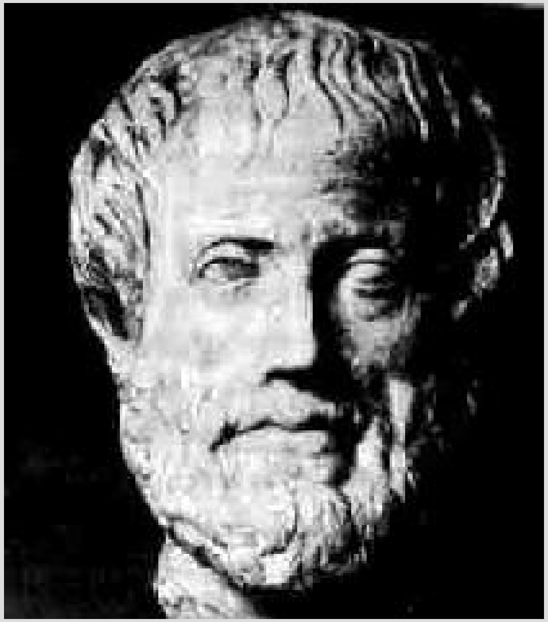
- 450B.C Stoics propositional Logic + inference (maybe)
- 322B.C Aristotle “syllogisms” inference rules



Zeno of Citium

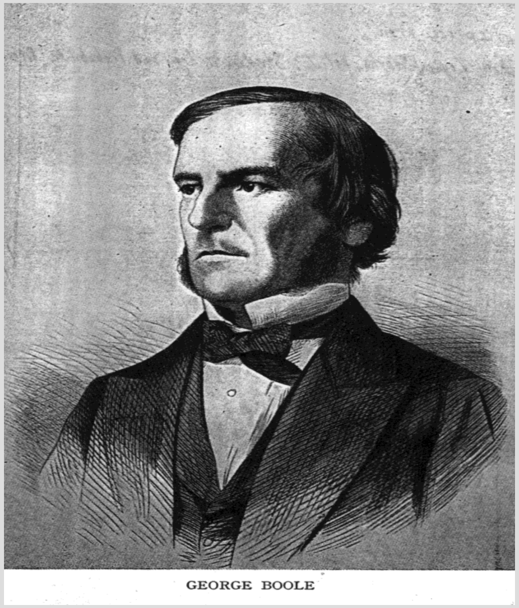
All knowledge
comes from the senses
mind is blank state

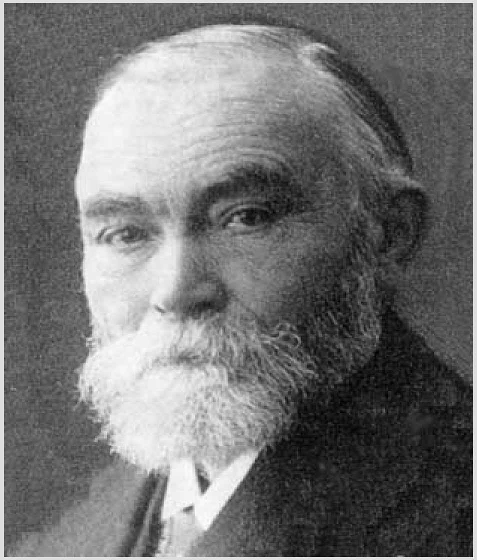
opposite of Plato's
idealism



Brief history of Reasoning

- 1565 Cardano Probability Theory (Propositional Logic + Uncertainty)
- 1847 Boole Propositional Logic
- 1879 Frege First-order Logic



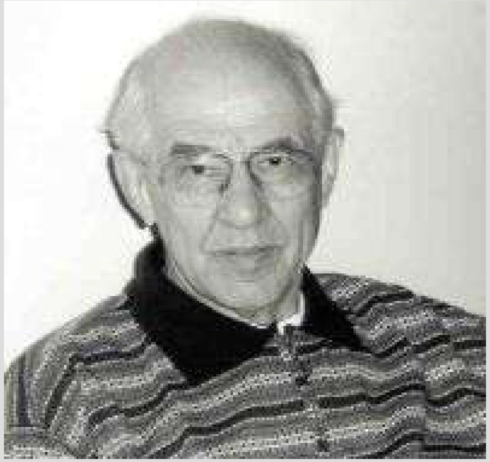


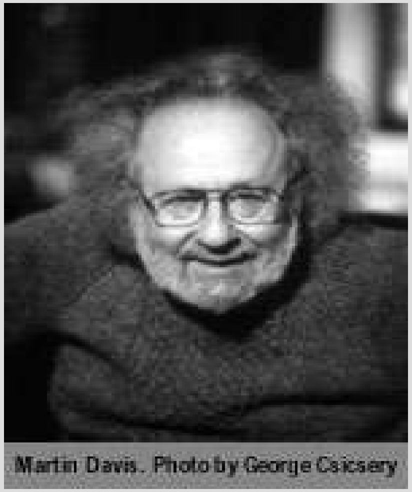
Brief History of Reasoning

- 1922 Wittgenstein Proof by truth tables
- 1930 Goedel ∃ complete algorithm for FOL
- 1930 Herbrand Complete algorithm for FOL (reduce to propositional)
- 1931 Goedel ¬ ∃ complete algorithm for arithmetic

Brief History of Reasoning

- 1960 – Davis/Putnam “practical” algorithm for propositional logic
- 1965 – Robinson “practical” algorithm for FOL resolution





Universal Instantiation (UI)

- Every instantiation of a universally quantified sentence is entailed by it:

$$\frac{\forall v \quad a}{\text{SUBST}(\{v/g\}, a)}$$

- e.g. $\forall$ x King(x) $\wedge$ Greedy(x) => Evil(x)
  - King(John) $\wedge$ Greedy(John)=>Evil(John)
  - King(Rich) $\wedge$ Greedy(Rich)=>Evil(Rich)
  - King(Father(John)) $\wedge$ Greedy(Father(John))=> Evil(Father(John))
  - ......

Existential Instantiation

- For any sentence a, variable v and constant symbol k that doesn't appear elsewhere in the KB:

$$\frac{\exists\ v \quad a}{\text{SUBST}(\{\ v/k\}\ ,\ a)}$$

- $\exists$ x Crown(x) $\wedge$ OnHead(x,John) yields
  - Crown(C1) $\wedge$ OnHead(C1,John) provided C1 is a new constant symbol, called Skolem constant

Converting FOL->PL

- UI can be applied several times to add new sentences; the new KB is equivalent with the old
- EI can be applied to once to replace the existential sentence; the new KB is not equivalent with the old but is satisfiable if the old is

Example

- $\forall$ x King(x) $\wedge$ Greedy(x) => Evil(x)
- King(John)
- Greedy(John)
- Brother(Richard, John)
- Instantiate the universal sentence in all possible ways
- What are the new propositional symbols ?

Reduction

- Claim: A ground sentence is entailed by new KB iff it is entailed by original FOL KB
- Claim: Every FOL KB can be propositionalized to preserve entailment
- Idea: propositionalize KB and query, apply resolution, return result
- Problem: with function symbols there are infinitely many ground terms:
  - e.g., Father(Father(Father(John)))

Herbrand Theorem

- 1930 – If a sentence a is entailed by an FOL KB it is entailed by a finite subset of the propositional KB
- Idea: For n = 0 to infinity do create propositional KB by instantiating with depth-n terms see if a is entailed by this KB
- Problem:
  - Works if a is entailed, loops forever if not
- Turing (1936), Church(1936)
  - Entailment in FOL is semidecidable

Problem with propositionalization

- Generates a lot of irrelevant sentences
  - $\forall$ x King(x) $\wedge$ Greedy(x) => Evil(x)
  - King(John)
  - $\forall$ y Greedy(y)
  - Brother(Richard, John)
  - It seems obvious that Evil(John) but propositionalization produces lots of facts such as Greedy(Richard) which are irrelevant

Unification

- We can get the inference immediately if we can find a substitution θ such that King(x) and Greedy(x) match King(John) and Greedy(y)
- $\Theta = \{x/John, y/John\}$ works
- Unify(a,b)=θ if $a\theta = b\theta$

Unification examples

P	Q	θ ?
Knows(John, x)	Knows(John, Jane)	
Knows(John, x)	Knows(y, Bob)	
Knows(John, x)	Knows(y, Mother(y))	
Knows(John, x)	Knows(x, Bob)	

Knows(John,x)	Knows(y,z)

Most general unifier Knows(John, z) rather than Knows(John, John)