CS 540 Introduction to Artificial Intelligence

Logic

University of Wisconsin-Madison

Spring 2025

Announcements

- **HW 1 online:**
  - Writing assignment---nothing too stressful
  - Deadline **today at 11:59PM**
- **HW 2**:
  - Probability
  - Deadline **Thursday Feb. 6th 11:59PM**

- Class roadmap:

Logic
NLP
Machine Learning: Introduction
Machine Learning: Unsupervised Learning I

Mostly Foundations

Logic & AI

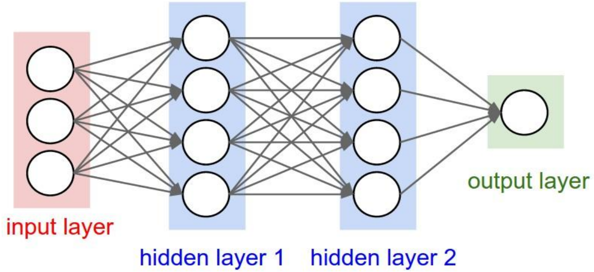
Why are we studying logic?

- **Traditional** approach to AI ('50s-'80s)
  - "Symbolic AI"
  - The Logic Theorist – 1956
    - Proved a bunch of theorems!
- Logic also the language of:
  - Knowledge rep., databases, etc.

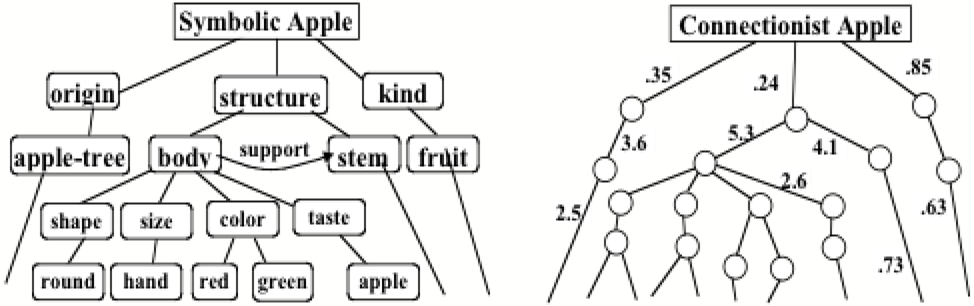
Symbolic vs Connectionist

Rival approach: **connectionist**

- Probabilistic models
- Neural networks
- **Extremely popular** last 20 years



Stanford CS231n

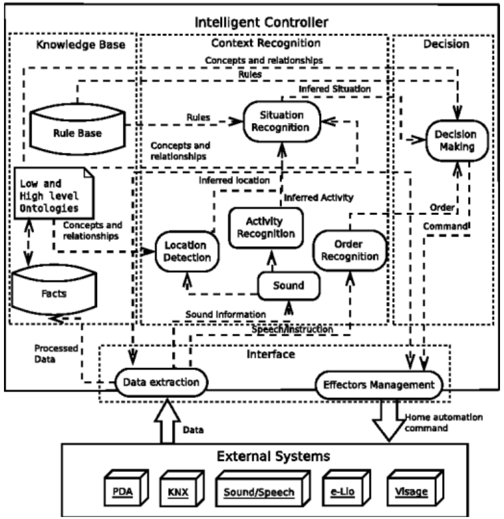


M. Minsky

Symbolic vs Connectionist

Which is better?

- Future: combination; best-of-both-worlds.
  - "Neurosymbolic AI"
  - **Example**: Markov Logic Networks



Outline

- Introduction to logic
  - Arguments, validity, soundness
- Propositional logic
  - Sentences, semantics, inference
- First order logic (FOL)
  - Predicates, objects, formulas, quantifiers


BEGRIFFSSCHRIFT,

EINE DER ARITHMETISCHEN NACHGEBILDETE

FORMELSPRACHE

DES REINEN DENKENS.

Dr. GOTTLOB FREGE

HALLE a/S.

VERLAG VON LOUIS NEBERT.

Basic Logic

- Arguments, premises, conclusions
  - Argument: a set of sentences (premises) + a sentence (a conclusion)
  - **Validity:** argument is valid iff it's necessary that if all premises are true, the conclusion is true
  - **Soundness:** argument is sound iff valid & premises true
  - **Entailment:** when valid arg., premises entail conclusion

Propositional Logic Basics

Logic Vocabulary:

- Sentences, symbols, connectives, parentheses
  - Symbols: P, Q, R, … (**atomic** sentences)
  - Connectives:

$\land$	and	[conjunction]
$\lor$	or	[disjunction]
$\Rightarrow$	implies	[implication]
$\Leftrightarrow$	is equivalent	[biconditional]
$\neg$	not	[negation]

  - Literal: P or negation ¬P

Propositional Logic Basics

Examples:

- $(P \lor Q) \Rightarrow S$
  - "If it is cold or it is raining, then I need a jacket"
- $Q \Rightarrow P$
  - "If it is raining, then it is cold"
- $\neg R$
  - "It is not hot"

Propositional Logic Basics

Several rules in place

- Precedence: $\neg$, $\wedge$, $\vee$, $\Rightarrow$, $\Leftrightarrow$
- Use parentheses when needed
- Sentences: **well-formed** or not well-formed:
  - $P \Rightarrow Q \Rightarrow S$ **not well-formed (not associative!)**

Sentences & Semantics

- Sentences: built up from symbols with connectives

  - **Interpretation**: assigning True / False to symbols (a row in truth table)
  - **Semantics**: interpretations for which sentence evaluates to True
  - **Model**: (of a set of sentences) interpretation for which all sentences are True

Another kind of model :)

Evaluating a Sentence

- Example:

P	Q	$\neg P$	$P \wedge Q$	$P \vee Q$	$P \Rightarrow Q$	$P \Leftrightarrow Q$
$false$	$false$	$true$	$false$	$false$	$true$	$true$
$false$	$true$	$true$	$false$	$true$	$true$	$false$
$true$	$false$	$false$	$false$	$true$	$false$	$false$
$true$	$true$	$false$	$true$	$true$	$true$	$true$

- Note:
  - If P is false, P $\Rightarrow$ Q is true regardless of Q ("5 is even implies 6 is odd" is True!)
  - Causality not needed: "5 is odd implies the Sun is a star" is True!)

Evaluating a Sentence: Truth Table

- **Ex**:

P	Q	R	¬P	Q∧R	¬P∨Q∧R	¬P∨Q∧R⇒Q
0	0	0	1	0	1	0
0	0	1	1	0	1	0
0	1	0	1	0	1	1
0	1	1	1	1	1	1
1	0	0	0	0	0	1
1	0	1	0	0	0	1
1	1	0	0	0	0	1
1	1	1	0	1	1	1

- **Satisfiable**
  - There exists some interpretation where the sentence is true.

Break & Quiz

Q 1.1: Suppose P is false, Q is true, and R is true. Does this assignment satisfy

(i) $\neg(\neg p \rightarrow \neg q) \wedge r$

(ii) $(\neg p \vee \neg q) \rightarrow (p \vee \neg r)$

- A. Both
- B. Neither
- C. Just (i)
- D. Just (ii)

Break & Quiz

Q 1.1: Suppose P is false, Q is true, and R is true. Does this assignment satisfy

(i) $\neg(\neg p \rightarrow \neg q) \land r$

(ii) $(\neg p \lor \neg q) \rightarrow (p \lor \neg r)$

- A. Both
- B. Neither
- **C. Just (i)**
- D. Just (ii)

Break & Quiz

Q 1.1: Suppose P is false, Q is true, and R is true. Does this assignment satisfy

(i) $\neg(\neg p \rightarrow \neg q) \wedge r$

(ii) $(\neg p \vee \neg q) \rightarrow (p \vee \neg r)$

- A. Both
- B. Neither
- **C. Just (i)**
- D. Just (ii)

Plug interpretation into each sentence.

For (i): $(\neg p \rightarrow \neg q)$ will be false so $\neg(\neg p \rightarrow \neg q)$ will be true and r is true by assignment.

For (ii): $(\neg p \vee \neg q)$ is true and $(p \vee \neg r)$ is false which makes the implication false.

Break & Quiz

Q 1.2: Let A =“Aldo is Italian” and B =“Bob is English”. Formalize “Aldo is Italian or if Aldo isn’t Italian then Bob is English”.

- a. $A \vee (\neg A \rightarrow B)$
- b. $A \vee B$
- c. $A \vee (A \rightarrow B)$
- d. $A \rightarrow B$

Break & Quiz

Q 1.2: Let A =“Aldo is Italian” and B =“Bob is English”. Formalize “Aldo is Italian or if Aldo isn’t Italian then Bob is English”.

- **a. $A \vee (\neg A \rightarrow B)$**
- **b. $A \vee B$ (equivalent!)**
- c. $A \vee (A \rightarrow B)$
- d. $A \rightarrow B$

Break & Quiz

Q 1.2: Let A ="Aldo is Italian" and B ="Bob is English". Formalize "Aldo is Italian or if Aldo isn't Italian then Bob is English".

- **a. $A \vee (\neg A \rightarrow B)$**
- **b. $A \vee B$ (equivalent!)**
- c. $A \vee (A \rightarrow B)$
- d. $A \rightarrow B$

Answer a. is the exact translation of the English sentence into a logic sentence. You can see that answer b. is also correct by writing out the truth table for all answers and seeing that a and b have the same truth tables.

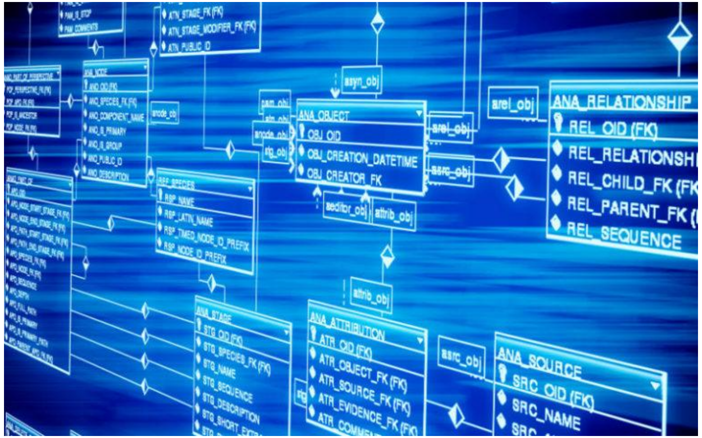
Or you can use the fact that $\neg A \rightarrow B = A \vee B$ and that $A \vee A \vee B = A \vee B$ to prove equivalence.

Knowledge Bases

- **Knowledge Base (KB)**: A set of sentences $\{A_1, \ldots, A_n\}$
  - Like a long sentence, connect with conjunction
  - KB: $A_1 \wedge A_2 \wedge \ldots \wedge A_n$

Model of a KB: interpretations where all sentences are True

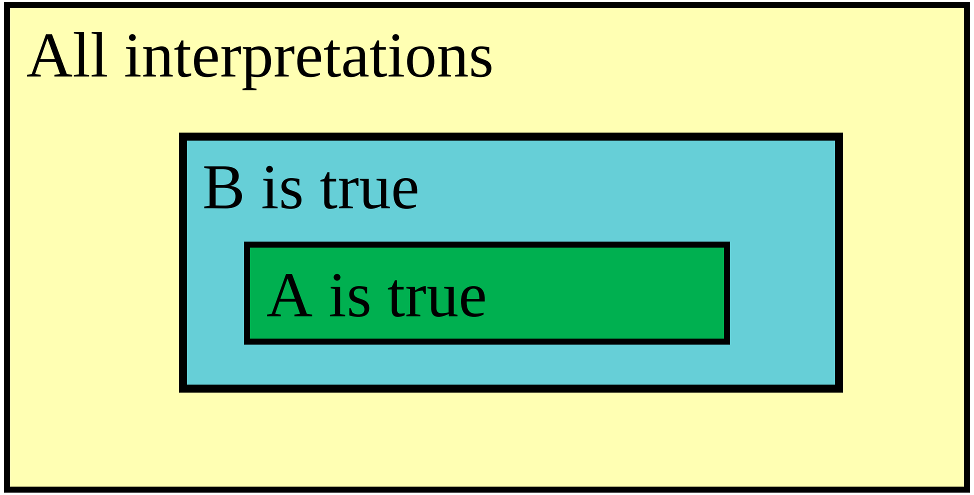
Goal: inference to discover new sentences



Entailment

Entailment: a sentence B logically follows from A

- Write A ⊨ B
- A ⊨ B iff in every interpretation where A is true, B is also true



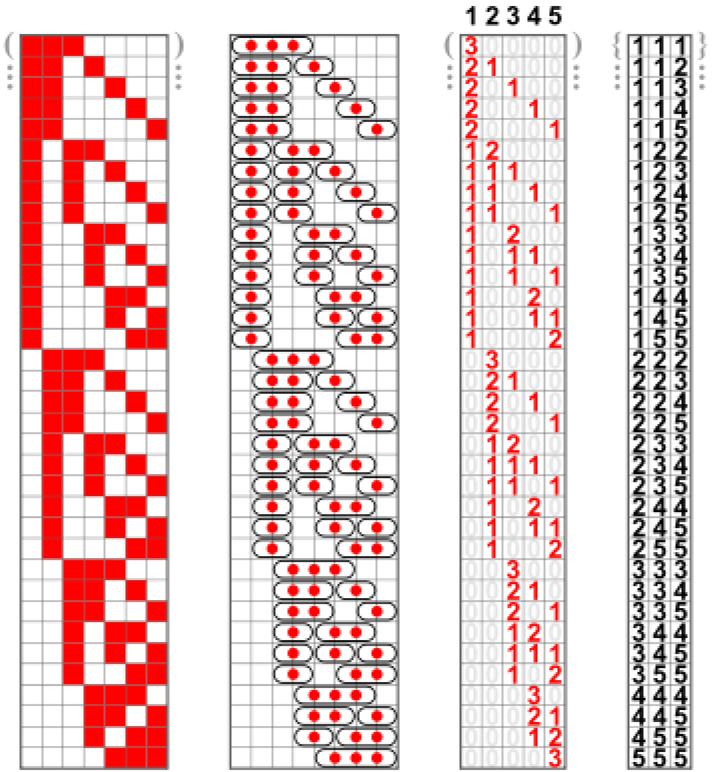
Inference

- Given a set of sentences (a KB), **logical inference** creates new sentences
  - Compare to prob. inference!
- **Challenges**:
  - Soundness
  - Completeness
  - Efficiency

Methods of Inference: **1. Enumeration**

- Enumerate all interpretations; look at the truth table
  - "Model checking"

- Downside: 2^n interpretations for n symbols



Wiki

Methods of Inference: **2. Using Rules**

- *Modus Ponens*: $(A \Rightarrow B, A) \vDash B$
- And-elimination
- Other rules on the next page
  - Commutativity, associativity, de Morgan's laws, distribution for conjunction/disjunction

Logical equivalences

$$(\alpha \land \beta) \equiv (\beta \land \alpha) \quad \text{commutativity of } \land$$
$$(\alpha \lor \beta) \equiv (\beta \lor \alpha) \quad \text{commutativity of } \lor$$
$$((\alpha \land \beta) \land \gamma) \equiv (\alpha \land (\beta \land \gamma)) \quad \text{associativity of } \land$$
$$((\alpha \lor \beta) \lor \gamma) \equiv (\alpha \lor (\beta \lor \gamma)) \quad \text{associativity of } \lor$$
$$\neg(\neg\alpha) \equiv \alpha \quad \text{double-negation elimination}$$
$$(\alpha \Rightarrow \beta) \equiv (\neg\beta \Rightarrow \neg\alpha) \quad \text{contraposition}$$
$$(\alpha \Rightarrow \beta) \equiv (\neg\alpha \lor \beta) \quad \text{implication elimination}$$
$$(\alpha \Leftrightarrow \beta) \equiv ((\alpha \Rightarrow \beta) \land (\beta \Rightarrow \alpha)) \quad \text{biconditional elimination}$$
$$\neg(\alpha \land \beta) \equiv (\neg\alpha \lor \neg\beta) \quad \text{de Morgan}$$
$$\neg(\alpha \lor \beta) \equiv (\neg\alpha \land \neg\beta) \quad \text{de Morgan}$$
$$(\alpha \land (\beta \lor \gamma)) \equiv ((\alpha \land \beta) \lor (\alpha \land \gamma)) \quad \text{distributivity of } \land \text{ over } \lor$$
$$(\alpha \lor (\beta \land \gamma)) \equiv ((\alpha \lor \beta) \land (\alpha \lor \gamma)) \quad \text{distributivity of } \lor \text{ over } \land$$

You can use these equivalences to modify sentences.

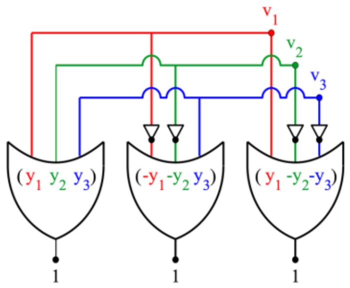
Methods of Inference: **3. Resolution**

- Convert to special form and use a single rule
- **Conjunctive Normal Form** (CNF)

$$\underbrace{(\neg A \vee B \vee C)}_{\text{a clause}} \wedge (\neg B \vee A) \wedge (\neg C \vee A)$$

Conjunction of clauses; each clause disjunction of literals

- Simple rules for converting to CNF



Conjunctive Normal Form (CNF)

$$(\neg B_{1,1} \lor P_{1,2} \lor P_{2,1}) \land (\neg P_{1,2} \lor B_{1,1}) \land (\neg P_{2,1} \lor B_{1,1})$$

- Replace all ⇔ using biconditional elimination
- Replace all ⇒ using implication elimination
- Move all negations inward using
  -double-negation elimination
  -de Morgan's rule
- Apply distributivity of ∨ over ∧

Convert example sentence into CNF

$B_{1,1} \Leftrightarrow (P_{1,2} \lor P_{2,1})$ starting sentence

$(B_{1,1} \Rightarrow (P_{1,2} \lor P_{2,1})) \land ((P_{1,2} \lor P_{2,1}) \Rightarrow B_{1,1})$

biconditional elimination

$(\neg B_{1,1} \lor P_{1,2} \lor P_{2,1}) \land (\neg(P_{1,2} \lor P_{2,1}) \lor B_{1,1})$

implication elimination

$(\neg B_{1,1} \lor P_{1,2} \lor P_{2,1}) \land ((\neg P_{1,2} \land \neg P_{2,1}) \lor B_{1,1})$

move negations inward

$(\neg B_{1,1} \lor P_{1,2} \lor P_{2,1}) \land (\neg P_{1,2} \lor B_{1,1}) \land (\neg P_{2,1} \lor B_{1,1})$

distribute $\lor$ over $\land$

Resolution Steps

- Given KB and β (query)
- Add $\neg \beta$ to KB, show this leads to empty (False. Proof by contradiction)
- Everything needs to be in CNF
- Example KB:
  - $B_{1,1} \Leftrightarrow (P_{1,2} \vee P_{2,1})$
  - $\neg B_{1,1}$
- Example query: $\neg P_{1,2}$

Resolution Preprocessing

- Add ¬ β to KB, convert to CNF:

  a1: $(\neg B_{1,1} \vee P_{1,2} \vee P_{2,1})$

  a2: $(\neg P_{1,2} \vee B_{1,1})$

  a3: $(\neg P_{2,1} \vee B_{1,1})$

  b: $\neg B_{1,1}$

  c: $P_{1,2}$
- Want to reach goal: *empty*

Resolution

- Take any two clauses where one contains some symbol, and the other contains its complement (negative)

  $P \vee Q \vee R$ $\qquad\qquad$ $\neg Q \vee S \vee T$

- Merge (resolve) them, throw away the symbol and its complement

  $P \vee R \vee S \vee T$

- If two clauses resolve and there's no symbol left, you have reached *empty* (False). KB |= β
- If no new clauses can be added, KB does not entail β

Resolution Example

a1: $(\neg B_{1,1} \lor P_{1,2} \lor P_{2,1})$

a2: $(\neg P_{1,2} \lor B_{1,1})$

a3: $(\neg P_{2,1} \lor B_{1,1})$

b: $\neg B_{1,1}$

c: $P_{1,2}$

Resolution Example

a1: $(\neg B_{1,1} \vee P_{1,2} \vee P_{2,1})$

a2: $(\neg P_{1,2} \vee B_{1,1})$

a3: $(\neg P_{2,1} \vee B_{1,1})$

b: $\neg B_{1,1}$

c: $P_{1,2}$

Step 1: resolve a2, c: $B_{1,1}$

Step 2: resolve above and b: *empty*

Break & Quiz

Q 2.1: Which has more rows: a truth table on *n* symbols, or a joint distribution table on *n* binary random variables?

- A. Truth table
- B. Distribution
- C. Same size
- D. It depends

Break & Quiz

Q 2.1: Which has more rows: a truth table on *n* symbols, or a joint distribution table on *n* binary random variables?

- A. Truth table
- B. Distribution
- **C. Same size**
- D. It depends

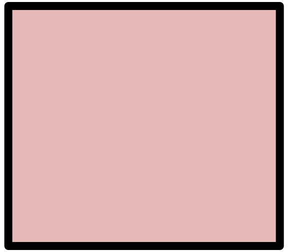
First Order Logic (FOL)

Propositional logic has some limitations

- Ex: how to say “all squares have four sides”
- No context, hard to generalize; express facts

FOL is a more expressive logic; works over

- Facts, Objects, Relations, Functions



First Order Logic Syntax

- **Term**: an object in the world
  - **Constant**: Alice, 2, Madison, Green, …
  - **Variables**: x, y, a, b, c, …
  - **Function**($term_1$, …, $term_n$)
    - Sqrt(9), Distance(Madison, Chicago)
    - Maps one or more objects to another object
    - Can refer to an unnamed object: LeftLeg(John)
    - Represents a user defined functional relation

- A **ground term** is a term without variables.
  - Constants or functions of constants

FOL Syntax

- **Atom**: smallest T/F expression
  - **Predicate**($term_1$, …, $term_n$)
    - Teacher(Jerry, you), Bigger(sqrt(2), x)
    - Convention: read “Jerry (is)Teacher(of) you”
    - Maps one or more objects to a truth value
    - Represents a user defined relation
  - **$term_1$ = $term_2$**
    - Radius(Earth)=6400km, 1=2
    - Represents the equality relation when two terms refer to the same object

FOL Syntax

- **Sentence**: T/F expression
  - Atom
  - Complex sentence using connectives: ∧ ∨ ¬ ⇒ ⇔
    - Less(x,22) ∧ Less(y,33)
  - Complex sentence using quantifiers ∀, ∃
- Sentences are evaluated under an interpretation
  - Which objects are referred to by constant symbols
  - Which objects are referred to by function symbols
  - What subsets defines the predicates

FOL Quantifiers

- Universal quantifier: ∀
- Sentence is true **for all** values of x in the domain of variable x.

- Main connective typically is ⇒
  - Forms if-then rules
  - "all humans are mammals"

$$\forall x \; \mathtt{human(x)} \Rightarrow \mathtt{mammal(x)}$$

  - Means if x is a human, then x is a mammal

FOL Quantifiers

- Existential quantifier: ∃
- Sentence is true **for some** value of x in the domain of variable x.

- Main connective typically is ∧
  - “some humans are male”

    $$\exists x\ \mathtt{human}(x) \wedge \mathtt{male}(x)$$

  - Means there is an x who is a human and is a male

Break & Quiz

Q 2.1: How many entries does a truth table have for a FOL sentence with k variables where each variable can take on n values?

- A. Truth tables are not applicable to FOL.
- B. 2^k
- C. n^k
- D. It depends

Break & Quiz

Q 2.1: How many entries does a truth table have for a FOL sentence with k variables where each variable can take on n values?

- A. Truth tables are not applicable to FOL.
- B. 2^k
- C. $\boldsymbol{n^k}$
- D. It depends

Break & Quiz

Q 2.1: How many entries does a truth table have for a FOL sentence with k variables where each variable can take on n values?

- A. Truth tables are not applicable to FOL.
- B. 2^k
- C. $\boldsymbol{n^k}$
- D. It depends

Must have one entry for every possible assignment of values to variables. That number is (C).