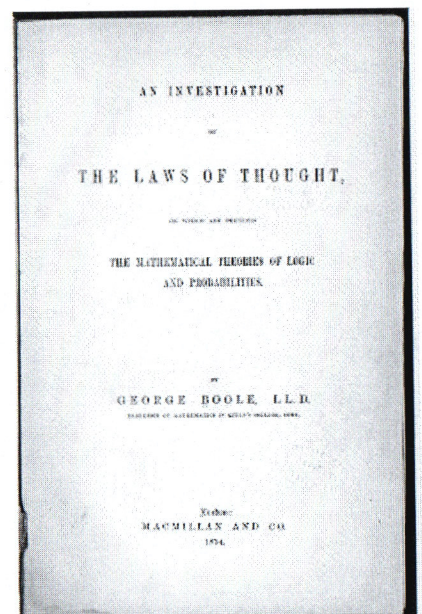
# CS 4100: Introduction to AI

Wayne Snyder
Northeastern University

---

Lecture 2: Introduction to Propositional Logic



AN INVESTIGATION

THE LAWS OF THOUGHT,

THE MATHEMATICAL THEORIES OF LOGIC
AND PROBABILITIES.

GEORGE BOOLE, LL.D.

MACMILLAN AND CO.
1854.

# Where did AI start?

**Recall one of our definitions of AI:** "[AI is] the theory and development of computer systems able to perform tasks that normally require human intelligence, such as visual perception, speech recognition, decision-making, and translation between languages."

This effort started in ancient times with the development of mathematics in Greece, India, and China. In the West, Aristotles' *Prior Analytics* had the most influence, introducing the ideas of

- **Propositions:** statements which are either true or false, and
- **Syllogisms:** basic patterns for reasoning, involving premises and a conclusion: the rule says that if the premises are true, then the conclusion must be true.

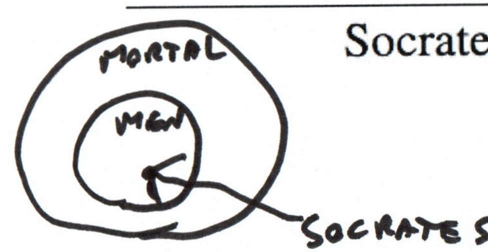
**Premise 1:** All men are mortal

**Premise 2:** Socrates is a man

THEREFORE,

**Conclusion:** Socrates is a man

$$\frac{\text{All men are mortal.} \qquad \text{Socrates is a man.}}{\text{Socrates is mortal.}}$$



# George Boole: The Laws of Thought

Fast forward to 1854, when George Boole published one of the foundational texts of modern mathematics, *The Laws of Thought*, described by Boole as follows:

> *"an investigation of the fundamental laws of those operations of the mind by which reasoning is performed; to give expression to them in the symbolical language of a calculus and, upon this foundation, to establish the science of logic and construct its method; to make that method itself the basis of a general method for the application of the mathematical doctrine of probabilities; and finally to collect from the various elements of truth brought to view in the course of these inquiries some probable intimations concerning the nature and constitution of the human mind."*

To express these propositions, let us assume—

$x$ = Motion began in time (and therefore)
$1 - x$ = Motion has existed from eternity.
$y$ = The first cause is an intelligent being.
$p$ = Motion has been eternally caused by some eternal intelligent being.
$q$ = Motion is self-existent.
$r$ = Motion has existed by endless successive communication.
$s$ = Matter is at rest.

The equations of the premises then are—

$$x = vy.$$
$$1 - x = v\{p(1-q)(1-r) + q(1-p)(1-r) + r(1-p)(1-q)\}.$$
$$p = vy.$$
$$q = vs(1-s) = 0.$$
$$r(1-q)(1-p) = 0.$$

Boole's system was based on simple mathematics:

1 = true
0 = false

(x and y) = x*y
(x or y) = x+y%2

(not x) = (1-x)

| X | Y | X AND Y |
|---|---|---|
| 1 | 1 | 1·1 = 1 |
| 1 | 0 | 1·0 = 0 |
| 0 | 1 | 0·1 = 0 |
| 0 | 0 | 0·0 = 0 |

# Propositional Logic: Syntax

**Definition 2.1** Let $Op = \{\neg, \land, \lor, \Rightarrow, \Leftrightarrow, (,)\}$ be the set of logical operators and $\Sigma$ a set of symbols. The sets $Op$, $\Sigma$ and $\{t, f\}$ are pairwise disjoint. $\Sigma$ is called the *signature* and its elements are the *proposition variables*. The set of propositional logic formulas is now recursively defined:

- $t$ and $f$ are (atomic) formulas.
- All proposition variables, that is all elements from $\Sigma$, are (atomic) formulas.
- If $A$ and $B$ are formulas, then $\neg A$, $(A)$, $A \land B$, $A \lor B$, $A \Rightarrow B$, $A \Leftrightarrow B$ are also formulas.

UNARY $\neg$

BINARY $\land, \lor, \Rightarrow, \Leftrightarrow$

PUNCTUATION: ( )

Precedence of Operators: $\neg, \land, \lor, \Longrightarrow, \Longleftrightarrow$.

$\Sigma = A, B, C, \ldots$

OR

$\Sigma = A_1, A_2, A_3, \ldots$

$A$ = TODAY IS SUNNY

$B$ = TODAY IS RAINING

$C$ = TODAY IS MONDAY

INDUCTIVE/RECURSIVE DEFINITION OF FORMULAE:

$A, B, (A \land B), \neg A$

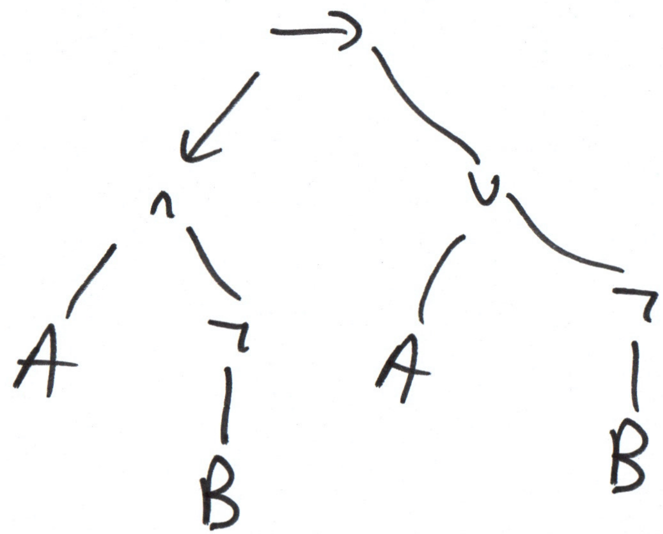
$A \Leftrightarrow B, \quad A \Rightarrow (A \Leftrightarrow B), \ldots$

$(A \land B)$ = TODAY IS SUNNY AND TODAY IS RAINING

$(C \Rightarrow A \lor B) \equiv$ ~~[illegible]~~ IF TODAY IS MONDAY THEN TODAY IS SUNNY OR TODAY IS RAINING

SYNTAX TREE FOR FORMULAE (BOOLEAN EXPRESSIONS):

$$(A \land \neg B) \to (A \lor \neg B)$$




SYMBOLS & RULES FOR COMBINATION

SYNTAX

COMPUTER LANGUAGES

SEMANTICS ↓

# Propositional Logic: Semantics = MEANING

**Definition 2.3** A mapping $I : \Sigma \to \{t, f\}$, which assigns a truth value to every proposition variable, is called an *interpretation*. (OR TRUTH ASSIGNMENT)

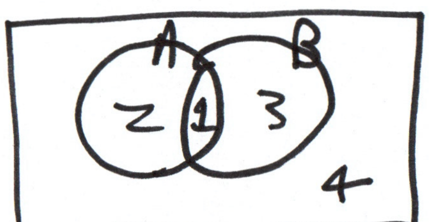
| | A | B |
|---|---|---|
| 1 | T | T |
| 2 | T | F |
| 3 | F | T |
| 4 | F | F |

AN INTERPRETATION IS A "STATE OF THE WORLD."

IF 2 SYMBOLS A AND B, THE WORLD CAN BE IN 4 DIFFERENT STATES.

A = TODAY IT'S RAINING

B = TODAY IT'S SUNNY




WITH N SYMBOLS WE CAN "SAY" $2^N$ DIFFERENT THINGS ABOUT THE WORLD.

# Propositional Logic: Semantics

Meaning of Operators:

NOTICE IS ONE EXCEPTION, OTHER 3 SAME

| $A$ | $B$ | $(A)$ | $\neg A$ | $A \wedge B$ | $A \vee B$ | $A \Rightarrow B$ | $A \Leftrightarrow B$ |
|---|---|---|---|---|---|---|---|
| $t$ | $t$ | $t$ | $f$ | $t$ | $t$ | $t$ | $t$ |
| $t$ | $f$ | $t$ | $f$ | $f$ | $t$ | $f$ | $f$ |
| $f$ | $t$ | $f$ | $t$ | $f$ | $t$ | $t$ | $f$ |
| $f$ | $f$ | $f$ | $t$ | $f$ | $f$ | $t$ | $t$ |

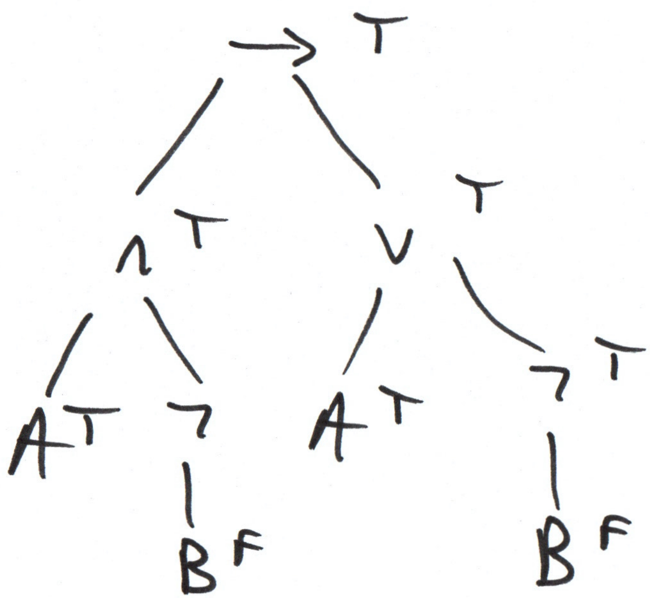
TO EVALUATE EXPRESSIONS, USE SAME TECHNIQUE:

$$(A \wedge \neg B) \rightarrow (A \vee \neg B)$$

| $A$ | $B$ | $\neg B$ | $(A \wedge \neg B)$ | $(A \vee \neg B)$ | $(A \wedge \neg B) \rightarrow (A \vee \neg B)$ |
|---|---|---|---|---|---|
| T | T | F | F | T | T |
| T | F | T | T | T | T |
| F | T | F | F | T | T |
| F | F | T | F | F | T |

# Propositional Logic: Semantics

CAN EVALUATE EXPRESSIONS BOTTOM UP (RECURSIVELY):




$A \mapsto T$

$B \mapsto F$

# Propositional Logic: Semantics of Formulae

**Definition 2.5** A formula is called

- *Satisfiable* if it is true for at least one interpretation.
- *Logically valid* or simply *valid* if it is true for all interpretations. True formulas are also called *tautologies*.
- *Unsatisfiable* if it is not true for any interpretation.

Every interpretation that satisfies a formula is called a *model* of the formula.

UNSATISFIABLE: $A \wedge \neg A$

SATISFIABLE: $A \vee \neg B$

TAUTOLOGY $(A \wedge \neg B) \rightarrow (A \vee \neg B)$

# Propositional Logic: Basic Notions

**Definition 2.4** Two formulas $F$ and $G$ are called semantically equivalent if they take on the same truth value for all interpretations. We write $F \equiv G$.

EXAMPLE:

| $A$ | $B$ | $A \to B$ | | $\neg A \lor B$ |
|---|---|---|---|---|
| T | T | T | | T |
| T | F | F | ← SAME → | F |
| F | T | T | | T |
| F | F | T | | T |

# Propositional Logic: Logical Equivalences

**Theorem 2.1** *The operations $\wedge$, $\vee$ are commutative and associative, and the following equivalences are generally valid:*

MAYBE $\equiv$ IS MORE APPROPRIATE

| | | | |
|---|---|---|---|
| $\neg A \vee B$ | $\Leftrightarrow$ | $A \Rightarrow B$ | *(implication)* |
| $A \Rightarrow B$ | $\Leftrightarrow$ | $\neg B \Rightarrow \neg A$ | *(contraposition)* |
| $(A \Rightarrow B) \wedge (B \Rightarrow A)$ | $\Leftrightarrow$ | $(A \Leftrightarrow B)$ | *(equivalence)* |
| $\neg(A \wedge B)$ | $\Leftrightarrow$ | $\neg A \vee \neg B$ | *(De Morgan's law)* |
| $\neg(A \vee B)$ | $\Leftrightarrow$ | $\neg A \wedge \neg B$ | |
| $A \vee (B \wedge C)$ | $\Leftrightarrow$ | $(A \vee B) \wedge (A \vee C)$ | *(distributive law)* |
| $A \wedge (B \vee C)$ | $\Leftrightarrow$ | $(A \wedge B) \vee (A \wedge C)$ | |
| $A \vee \neg A$ | $\Leftrightarrow$ | $w$ | *(tautology)* |
| $A \wedge \neg A$ | $\Leftrightarrow$ | $f$ | *(contradiction)* |
| $A \vee f$ | $\Leftrightarrow$ | $A$ | |
| $A \vee w$ | $\Leftrightarrow$ | $w$ | |
| $A \wedge f$ | $\Leftrightarrow$ | $f$ | |
| $A \wedge w$ | $\Leftrightarrow$ | $A$ | |

THESE ARE USED TO CONVERT TO CNF

ALSO: $A \vee B \equiv B \vee A$

$A \wedge B \equiv B \wedge A$

| $A$ | $B$ | $\neg A$ | $\neg B$ | $A \to B$ | $\neg B \to \neg A$ |
|---|---|---|---|---|---|
| T | T | F | F | T | T |
| T | F | F | T | T | T |
| F | T | T | F | F | F |
| F | F | T | T | T | T |

← SAME →

# Propositional Logic: Basic Notions

**Definition 2.6** A formula *KB entails* a formula $Q$ (or $Q$ *follows* from $KB$) if every model of $KB$ is also a model of $Q$. We write $KB \models Q$.

If a formula Q is valid (a tautology), then we write $\models Q$.

KB IS FORMULA OR SET OF FORMULAE (CONSIDERED AS A CONJUNCTION)

I SATISFIES SET OF FMLA IF SATISFIES EVERY MEMBER.

$$\{A, A \to B, B\} \equiv A \wedge (A \to B) \wedge B$$

SO EMPTY SET IS A TAUTOLOGY (ANY INTERPRETATION SATISFIES EVERY MEMBER OF SET)

$\models Q$ SAME AS $\emptyset \models Q$

# Propositional Logic: Basic Notions

**Theorem 2.2** (Deduktionstheorem)

$$A \models B \text{ if and only if } \models A \Rightarrow B.$$

$A \models B \Rightarrow \models A \Rightarrow B$

2 CASES (1) I SATISFIES $A$

THEN I SATISFIES $B$ AND

I SATISFIES $A \Rightarrow B$ $(T \to T)$

(2) I DOESN'T SATISFY $A$

THEN I SATISFIES $A \Rightarrow B$

$\Leftarrow$ IS VERY SIMILAR

# Propositional Logic: Proof Calculi

A proof calculus is a collection of syllogisms (rules) for deriving consequences of a collection of formulae. It is an entirely syntactic procedure.

If we can derive a formula Q from a formula (or set of formulae) KB, we write

$$KB \vdash Q$$

and we say that *KB derives Q*.

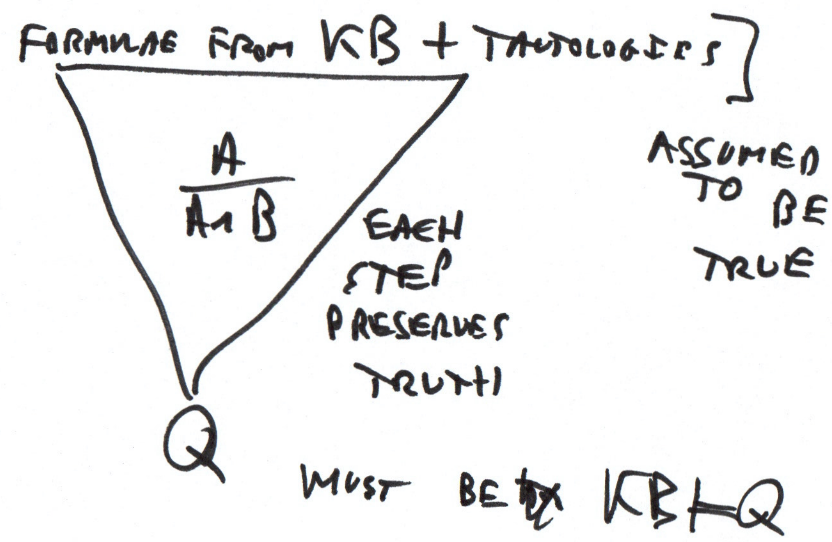
A large number of different proof calculi have been developed, here is a brief sample....

SYLLOGISMS

$$\frac{A \wedge B}{A} \qquad \frac{\neg\neg A}{A} \qquad \frac{A \to B}{\neg A \vee B}$$

$$\frac{A}{A \vee B} \qquad \frac{A}{\neg\neg A} \qquad \frac{\neg A \vee B}{A \to B}$$

A PROOF IS A TREE OF SUCH STEPS



# Propositional Logic: Proof Calculi

EXAMPLE 1

$(A \wedge B) \vdash (A \vee B)$

$$\frac{\dfrac{(A \wedge B)}{A}}{A \vee B}$$

✓

EXAMPLE 2

$KB = \{A \to B, B \to C, A\}$

$Q = C$

$$\frac{\dfrac{A \quad A \to B}{B} \quad B \to C}{C}$$

MODUS PONENS:

$$\frac{A \quad A \to B}{B}$$

# Propositional Logic: Proof Calculi

Unfortunately, these are not always easy to understand (for humans) and very difficult to generate by machine! They generally require some creativity and insight into the problem.

$$
\begin{prooftree}
\AxiomC{}
\RightLabel{$(I)$}
\UnaryInfC{$A \vdash A$}
\RightLabel{$(\neg R)$}
\UnaryInfC{$\vdash \neg A, A$}
\RightLabel{$(PR)$}
\UnaryInfC{$\vdash A, \neg A$}
\AxiomC{}
\RightLabel{$(I)$}
\UnaryInfC{$B \vdash B$}
\AxiomC{}
\RightLabel{$(I)$}
\UnaryInfC{$C \vdash C$}
\RightLabel{$(\vee L)$}
\BinaryInfC{$B \vee C \vdash B, C$}
\RightLabel{$(PR)$}
\UnaryInfC{$B \vee C \vdash C, B$}
\RightLabel{$(\neg L)$}
\UnaryInfC{$B \vee C, \neg C \vdash B$}
\AxiomC{}
\RightLabel{$(I)$}
\UnaryInfC{$\neg A \vdash \neg A$}
\RightLabel{$(\rightarrow L)$}
\BinaryInfC{$(B \vee C), \neg C, (B \rightarrow \neg A) \vdash \neg A$}
\RightLabel{$(\wedge L_1)$}
\UnaryInfC{$(B \vee C), \neg C, ((B \rightarrow \neg A) \wedge \neg C) \vdash \neg A$}
\RightLabel{$(PL)$}
\UnaryInfC{$(B \vee C), ((B \rightarrow \neg A) \wedge \neg C), \neg C \vdash \neg A$}
\RightLabel{$(\wedge L_2)$}
\UnaryInfC{$(B \vee C), ((B \rightarrow \neg A) \wedge \neg C), ((B \rightarrow \neg A) \wedge \neg C) \vdash \neg A$}
\RightLabel{$(CL)$}
\UnaryInfC{$(B \vee C), ((B \rightarrow \neg A) \wedge \neg C) \vdash \neg A$}
\RightLabel{$(PL)$}
\UnaryInfC{$((B \rightarrow \neg A) \wedge \neg C), (B \vee C) \vdash \neg A$}
\RightLabel{$(\rightarrow L)$}
\BinaryInfC{$((B \rightarrow \neg A) \wedge \neg C), (A \rightarrow (B \vee C)) \vdash \neg A, \neg A$}
\RightLabel{$(CR)$}
\UnaryInfC{$((B \rightarrow \neg A) \wedge \neg C), (A \rightarrow (B \vee C)) \vdash \neg A$}
\RightLabel{$(PL)$}
\UnaryInfC{$(A \rightarrow (B \vee C)), ((B \rightarrow \neg A) \wedge \neg C) \vdash \neg A$}
\RightLabel{$(\rightarrow R)$}
\UnaryInfC{$(A \rightarrow (B \vee C)) \vdash (((B \rightarrow \neg A) \wedge \neg C) \rightarrow \neg A)$}
\RightLabel{$(\rightarrow R)$}
\UnaryInfC{$\vdash ((A \rightarrow (B \vee C)) \rightarrow (((B \rightarrow \neg A) \wedge \neg C) \rightarrow \neg A))$}
\end{prooftree}
$$

UGH!

# Propositional Logic: Syntax and Semantics

(CORRECT)

**Definition 2.7** A calculus is called *sound* if every derived proposition follows semantically. That is, if it holds for formulas $KB$ and $Q$ that

$$\text{if} \quad KB \vdash Q \quad \text{then} \quad KB \models Q.$$

A calculus is called *complete* if all semantic consequences can be derived. That is, for formulas $KB$ and $Q$ the following holds:

$$\text{if} \quad \underbrace{KB \models Q}_{\text{SEMANTIC}} \quad \text{then} \quad \underbrace{KB \vdash Q}_{\text{SYNTAX}}.$$

SO

$$KB \models Q \iff KB \vdash Q$$

EQUIVALENT!

$\models$ IS HARD TO CHECK
(EXPONENTIAL # OF INTERPRETATIONS)
SO WE FOCUS ON $\vdash$

# Propositional Logic: Syntax and Semantics

$$\begin{tikzcd}
KB \arrow[r, "\vdash", "\text{derivation}"'] \arrow[d, "\text{interpretation}"'] & Q \arrow[d, "\text{interpretation}"] \\
\mathrm{Mod}(KB) \arrow[r, "\models", "\text{entailment}"'] & \mathrm{Mod}(Q)
\end{tikzcd}$$

syntactic level
(formula)

semantic level
(interpretation)

$\mathrm{Mod}(KB) \subseteq \mathrm{Mod}(Q)$

# Propositional Logic: Automated Theorem Proving

An alternative proof calculus that is more suitable for machine implementation is the Resolution Method, invented by Abraham Robinson in 1965. It is a method of proof by contradition or refutation:

**Theorem 2.3** (Proof by contradiction) $KB \models Q$ *if and only if* $KB \wedge \neg Q$ *is unsatisfiable.*

SEE NEXT PAGE

The Resolution Rule is a generalization of Modus Ponens and similar rules that explain how implication can be used:

$$\frac{A \quad A \Rightarrow B}{B} \qquad \frac{A \quad \neg A \vee B}{B}$$

$$\frac{A \Rightarrow B \quad B \Rightarrow C}{A \Rightarrow C} \qquad \frac{\neg A \vee B \quad \neg B \vee C}{\neg A \vee C}$$

THM
$KB \models Q \iff KB \cup \{\neg Q\}$ IS UNSATISFIABLE

PROOF $(\Longrightarrow)$
2 CASES

I SATISFIES KB

$\Rightarrow$ I SATISFIES Q

$\Rightarrow$ I DOESN'T SATISFY $\neg Q$

$\Rightarrow$ " " " $KB \cup \{\neg Q\}$

I DOESN'T SAT. KB

$\Rightarrow$ I DOESN'T SAT. $KB \cup \{\neg Q\}$

$(\Longleftarrow)$ IS SIMILAR