Great Place To Work® Certified FEB 2023 – FEB 2024 INDIA

UPES

UNIVERSITY OF TOMORROW

nirf RANKING 2023

#RANKED 52 IN INDIA

#University Category

QS WORLD UNIVERSITY RANKINGS

NO.1 PVT. UNIVERSITY IN ACADEMIC REPUTATION IN INDIA

ACCREDITED WITH GRADE A NAAC

ACCREDITED **GRADE 'A'** BY NAAC

E-LEAD

PERFECT SCORE OF **150/150** AS A TESTAMENT TO EXCEPTIONAL E-LEARNING METHODS

Unit 2 : Logic for AI

UPES
UNIVERSITY OF TOMORROW

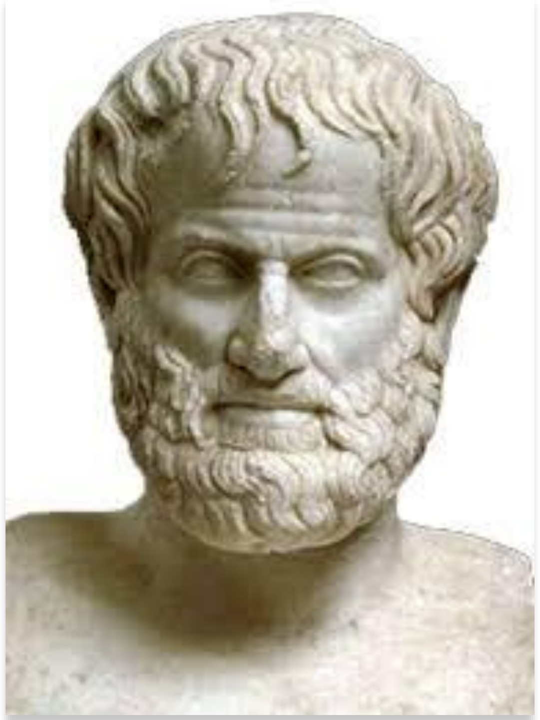
Table of Contents

UPES
UNIVERSITY OF TOMORROW

Introduction to Logic

What is Logic?
The study of principles and methods used to distinguish **correct** reasoning from **incorrect** reasoning.

Father of Logic
Aristotle (384–322 BCE): Often considered the "father of logic," **Aristotle** developed the first formal system of logic, known as **syllogistic logic**. His work, particularly in "Organon," laid the groundwork for future logical studies.



Propositional Logic

- Propositional logic is also known as **propositional calculus** or **sentential logic**.
- It is a branch of logic that deals with **propositions** and their relationships through logical **connectives**.
- It forms the foundation for many areas of formal logic, computer science, and mathematics.

UPES
UNIVERSITY OF TOMORROW

Propositions

- A proposition is a declarative statement that is either **True** or **False** but not both.
- Examples include:
  - "It is raining" (True or False)
  - "2 + 2 = 4" (True).

UPES
UNIVERSITY OF TOMORROW

Logical Connectives

Connectives: Symbols representing logical operations.

Types of Connectives:

- *Negation (¬) :* The negation of a proposition P is denoted as ¬P and is True if P is False, and False if P is True.
- *Conjunction (∧) :* The conjunction of propositions P and Q is denoted as P∧Q and is True if both P and Q are True.
- *Disjunction (∨) :* The disjunction of propositions P and Q is denoted as P∨Q and is True any one of P or Q is True.
- *Implication (→) :* The implication P→Q is true if whenever P is True, Q is also True. It is False only if P is True and Q is False.
- *Bi-implication (↔) :* The biconditional P↔Q is true if P and Q are either both True or both False.

UPES
UNIVERSITY OF TOMORROW

Implication ($P \rightarrow Q$)

Meaning: *If P, then Q*

Rule: False **only** when P is True and Q is False.

P	Q	P → Q
T	T	T
T	F	F
F	T	T
F	F	T

Example: P: “It is raining.” Q: “The ground is wet.”

If it is raining (P = True), then the ground should be wet (Q = True) → implication is True. If it rains (P = True) but the ground is not wet (Q = False) → implication is False. If it’s not raining (P = False), the statement “If it rains, then the ground will be wet” is still considered True, regardless of Q.

Bi-implication (P ↔ Q)

- Meaning: P if and only if Q
- Rule: True when P and Q have the same truth value

(both True or both False).

P	Q	P ↔ Q
T	T	T
T	F	F
F	T	F
F	F	T

Example:

- P: "You can enter the lab."
- Q: "You have an ID card."
- The rule says: *You can enter the lab if and only if you have an ID card.*
  - If you have an ID card (Q = True), you can enter (P = True) → True.
  - If you don't have an ID card (Q = False), you cannot enter (P = False) → True.
  - If you have an ID card but still cannot enter → False.
  - If you don't have an ID card but can still enter → False.

Truth Tables

Purpose: Systematically determine the truth value of compound propositions.

Construction: Assign truth values to variables and apply logical connectives.

Example: Truth table for $(p \wedge q) \vee \neg p$

Truth Table Examples

P	$\neg P$
T	F
F	T

P	Q	$P \vee Q$
T	T	T
T	F	T
F	T	T
F	F	F

P	Q	$P \wedge Q$
T	T	T
T	F	F
F	T	F
F	F	F

P	Q	$P \to Q$
T	T	T
T	F	F
F	T	T
F	F	T

P	Q	$P \leftrightarrow Q$
T	T	T
T	F	F
F	T	F
F	F	T

For a detailed tutorial on Truth Tables, please refer to:
https://sites.millersville.edu/bikenaga/math-proof/truth-tables/truth-tables.html

Laws of Propositional Logic

- **Commutative Laws**: $p \wedge q \equiv q \wedge p$
- **Associative Laws**: $(p \wedge q) \wedge r \equiv p \wedge (q \wedge r)$
- **Distributive Laws**: $p \wedge (q \vee r) \equiv (p \wedge q) \vee (p \wedge r)$
- **De Morgan's Laws**: $\neg(p \wedge q) \equiv \neg p \vee \neg q$
- **Idempotent Laws**: $p \wedge p \equiv p$

Example: Applying De Morgan's Law: $\neg(p \vee q) \equiv \neg p \wedge \neg q$

UPES
UNIVERSITY OF TOMORROW

Examples

Simplifying the expression $\neg(p \lor q) \land (p \to q)$

1. Apply De Morgan's Law to $\neg(p \lor q)$:

   $\neg(p \lor q) \equiv \neg p \land \neg q$

2. Apply the definition of implication to $p \to q$:

   $p \to q \equiv \neg p \lor q$

3. Substitute these into the original expression:

   $(\neg p \land \neg q) \land (\neg p \lor q)$

4. Distribute $\neg p$ across $(\neg p \land \neg q)$ and $(\neg p \lor q)$:

   $(\neg p \land \neg q) \land (\neg p \lor q) \equiv (\neg p \land \neg q \land \neg p) \lor (\neg p \land \neg q \land q)$

5. Simplify the expression:

   $(\neg p \land \neg q \land \neg p) \lor (\neg p \land \neg q \land q) \equiv \neg p \land \neg q$

So, the simplified expression is:

$\neg p \land \neg q$

Tautology, Contradiction and Contingency

- **Tautology**

  A proposition that is **always true**, no matter what the truth values of its components are.

  Symbolically: A formula is a tautology if its truth table column is always **T**.

- **Contradiction**

  A proposition that is **always false**, regardless of the truth values of its components.

  Symbolically: A formula is a contradiction if its truth table column is always **F**.

- **Contingency**

  A proposition that is **sometimes true, sometimes false** (i.e., not a tautology and not a contradiction).

  It depends on the truth values of its variables.

Practice Questions:

Practice: Classify each formula as Tautology, Contradiction, or Contingency

1. $P \vee \neg P$
2. $P \wedge \neg P$
3. $(P \rightarrow Q) \vee (Q \rightarrow P)$
4. $(P \vee Q) \wedge (\neg P \vee Q)$
5. $(P \rightarrow Q) \wedge (Q \rightarrow P)$
6. $(P \vee Q) \rightarrow (P \wedge Q)$

UPES
UNIVERSITY OF TOMORROW

A. Truth Table Construction

1. Construct the truth table for the following propositions:

   a) $\neg P \vee Q$

   b) $(P \wedge Q) \rightarrow R$

   c) $(P \vee Q) \wedge (\neg P \vee R)$

   d) $(P \rightarrow Q) \leftrightarrow (\neg Q \rightarrow \neg P)$

B. Determine Validity / Equivalence

2. Show whether the following propositions are **logically equivalent:**

   a) $P \rightarrow Q$ and $\neg P \vee Q$

   b) $\neg(P \wedge Q)$ and $\neg P \vee \neg Q$ (De Morgan's Law)

   c) $(P \vee Q) \rightarrow R$ and $(P \rightarrow R) \wedge (Q \rightarrow R)$

UPES UNIVERSITY OF TOMORROW

What is Predicate Logic?
First order predicate logic (FOPL)

Predicate Logic: Deals with predicates and quantified variables.

Predicates: Statements containing variables that can be either true or false.

Quantifiers: Symbols indicating the extent of the variable's applicability (e.g., ∀ for universal quantification, ∃ for existential quantification).

Example: $\forall x(P(x))$ represents "for all x, P(x)".

UPES
UNIVERSITY OF TOMORROW

Predicates and Quantifiers

Predicates: Expressions containing variables that can be true or false.

Quantifiers: Determine the scope of variables.

Examples:

Predicate: $P(x)$ - "x is a prime number."

Quantified Statement: $\forall x(P(x))$ - "All x are prime numbers."

Quantifier Negation

Negating Quantified Statements:
Universal Quantification: $\neg(\forall x(P(x))) \equiv \exists x(\neg P(x))$
Existential Quantification: $\neg(\exists x(P(x))) \equiv \forall x(\neg P(x))$

Example:

Negating "All cats are mammals"
becomes
"Some cats are not mammals".

Role of Predicate Logic in AI

- **Knowledge Representation:** It provides a structure for representing complex facts about objects and their relationships in a system.
- **Reasoning:** AI systems use predicate logic to **infer new information** from existing facts, making it suitable for **decision-making tasks**.

- **Example:**

"John is the father of Mary" can be represented as: Father(John, Mary)

This expression tells us that there is a relationship (Father) between two objects (John and Mary). Predicate logic enables us to model and reason about such relationships effectively.

Rules of Inference

- **Modus Ponens (Implication Elimination):** If $P \rightarrow Q$ and P is true, then Q is true.
  Example: If patient has fever → patient is sick. Patient has fever. ∴ Patient is sick.

- **Modus Tollens:** If $P \rightarrow Q$ and $\neg Q$, then $\neg P$.
  Example: If it rains → ground is wet. Ground is not wet. ∴ It did not rain.

- **Hypothetical Syllogism:** If $P \rightarrow Q$ and $Q \rightarrow R$, then $P \rightarrow R$.
  Example: If it rains → streets are wet. If streets are wet → traffic slows. ∴ If it rains → traffic slows.

- **Disjunctive Syllogism:** If $P \lor Q$ and $\neg P$, then Q.
  Example: Either it is sunny or cloudy. It is not sunny. ∴ It is cloudy.

- **Resolution (used in AI & Automated Theorem Proving):** From $(P \lor Q)$ and $(\neg P \lor R)$, infer $(Q \lor R)$. This is the basis of many logic programming languages (e.g., Prolog) and automated reasoning systems.

UPES
UNIVERSITY OF TOMORROW

Predicate Logic Inference Rules

- **Universal Instantiation:** If $\forall x(P(x))$ is true, then $P(a)$ is true for any individual a.
- **Existential Instantiation:** If $\exists x(P(x))$ is true, then $P(a)$ is true for some particular a.
- **Universal Generalization:** If $P(a)$ is true for any individual a, then $\forall x(P(x))$ is true.
- **Existential Generalization:** If $P(a)$ is true for some particular a, then $\exists x(P(x))$ is true.

Example: Using Universal Instantiation: If "All cats are mammals", then "Fluffy is a mammal".

UPES
UNIVERSITY OF TOMORROW

Predicate Logic Equivalences

- **Universal Quantifier Equivalence**:
  $\neg\forall x(P(x))\equiv\exists x(\neg P(x))$

- **Existential Quantifier Equivalence**:
  $\neg\exists x(P(x))\equiv\forall x(\neg P(x))$

- **Negation Equivalence**:
  $\neg(\forall x(P(x)))\equiv\exists x(\neg P(x))$ and $\neg(\exists x(P(x)))\equiv\forall x(\neg P(x))$

UPES
UNIVERSITY OF TOMORROW

Applications of Logic

Mathematics: Proving theorems, solving equations.

Computer Science: Designing algorithms, programming languages.

Philosophy: Analyzing arguments, studying language.

Linguistics: Understanding sentence structure, semantics.

Example: Using logic to verify software correctness.

Rule of Inference	Form	Tautology	Description
Modus Ponens (MP)	If $p \rightarrow q$ and p, then q.	$p \land (p \rightarrow q)) \rightarrow \neg q$	If P implies Q, and P is true, then Q is true.
Modus Tollens (MT)	If $p \rightarrow q$ and $\neg q$, then $\neg p$.	$(\neg q \land (p \rightarrow q)) \rightarrow \neg p$	If P implies Q, and Q is false, then P is false.
Hypothetical Syllogism (HS)	If $p \rightarrow q$ and $q \rightarrow r$, then $p \rightarrow r$.	$((p \rightarrow q) \land (q \rightarrow r)) \rightarrow (p \rightarrow r)$	If P implies Q and Q implies R, then P implies R.
Disjunctive Syllogism (DS)	If $p \lor q$, and $\neg p$, then q.	$(\neg p \land (p \lor q)) \rightarrow q$	If P or Q is true, and P is false, then Q is true.
Conjunction (Conj)	If p and q, then $p \land q$.	$(p \land q) \rightarrow (p \land q)$ or $p \rightarrow (q \rightarrow (p \land q))$	If P and Q are true, then P and Q are true.
Simplification (Simp)	If $p \land q$, then p	$(p \land q) \rightarrow p$	If P and Q are true, then P is true
Addition (Add)	If p, then $p \lor q$	$p \rightarrow (p \lor q)$	If P is true, then P or Q is true.
Absorption(Abs)	If $p \rightarrow q$, then $p \rightarrow (p \land q)$	$(p \rightarrow q) \rightarrow (p \rightarrow (p \land q))$	If P implies Q, then P implies P or Q is true.
Resolution	If $p \lor q$, and $\neg p \lor r$, then $q \lor r$.	$p \lor q, \neg p \lor r \Rightarrow q \lor r$	If P or Q is true, and not P or R is true, then Q or R is true.

UPES
UNIVERSITY OF TOMORROW

What is Resolution?

- Resolution Method is an inference rule which is used in both proposition as well as First order predicate logic (FOPL) in different ways.

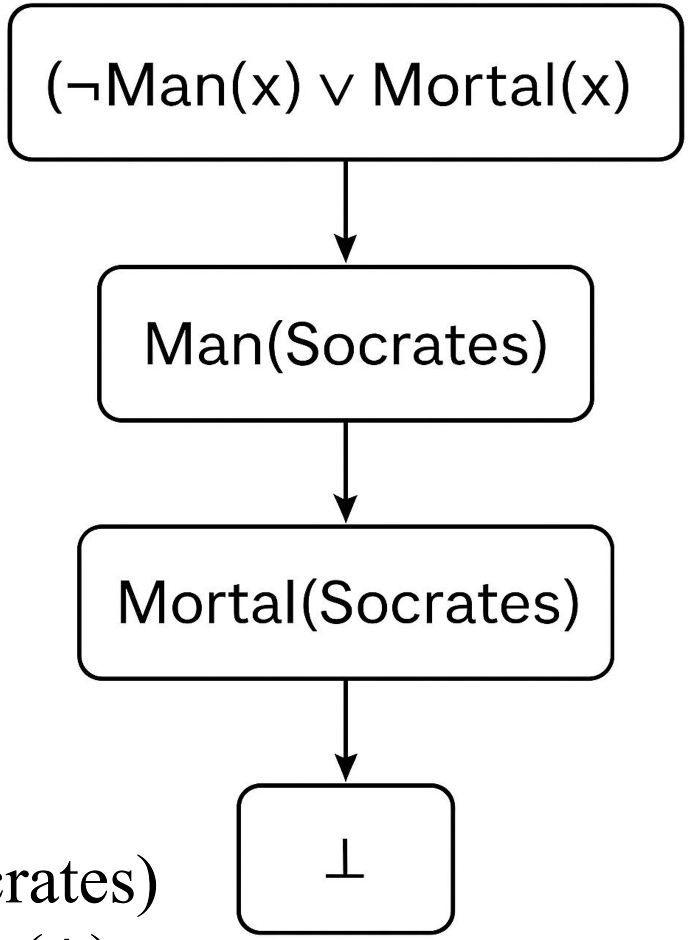
- This method is basically used for proving the satisfiability of a sentence.

- In resolution method, we use "**Proof by Refutation**" technique to prove the given statement.

- Proof by Refutation is a method of proving a statement S by showing that assuming the opposite ($\neg S$) leads to a contradiction. If $\neg S$ leads to something false ($\perp$), then S must be true.

- If you have two sentences:
  $(P \vee A)$ → means "P or A is true."
  $(\neg P \vee B)$ → means "not P or B is true."
- You can **resolve** them by canceling out **P and ¬P**, giving:
  $(A \vee B)$
- **Example (Proving Socrates is Mortal)**
  Knowledge base:
  All men are mortal = (¬Man(x) ∨ Mortal(x))
  Socrates is a man = (Man(Socrates))
  Assume the opposite:
  Socrates is not mortal = (¬Mortal(Socrates))
  Steps:
  From (¬Man(x) ∨ Mortal(x)) and Man(Socrates) → Mortal(Socrates)
  From Mortal(Socrates) and ¬Mortal(Socrates) → Contradiction (⊥)
  So, Socrates must be mortal.

Resolution



Knowledge Base (facts & rules):

If it is sunny, then John is happy.
→ (¬Sunny ∨ Happy(John))

It is sunny.
→ Sunny

We want to prove: **Happy(John)**

- **Step 1: Negate the goal**

  We want to prove Happy(John), so we assume the opposite: ¬Happy(John)

- **Step 2: Write all clauses**
  - (¬Sunny ∨ Happy(John))
  - Sunny
  - ¬Happy(John)

- **Step 3: Apply Resolution**

  From (¬Sunny ∨ Happy(John)) and Sunny → **Happy(John)**

  From Happy(John) and ¬Happy(John) → ⊥ **Contradiction**

- **Step 4: Conclusion**

  Since the negation (¬Happy(John)) caused a contradiction, we conclude : **Happy(John) is true.**

UPES
UNIVERSITY OF TOMORROW

Resolution Method in AI

- There are two methods for resolution in AI:
  - **Resolution Method in Proposition logic**
  - **Resolution Method in Predicate logic**

UPES
UNIVERSITY OF TOMORROW

Resolution Method in Propositional Logic

- Resolution in Propositional Logic: Works with simple true/false statements (no variables, no quantifiers).
- Example: Knowledge Base:
  - $P \vee Q$P∨Q (P or Q)
  - $\neg P \vee R$¬P∨R (not P or R)
- Resolution Rule:

  From $(P \vee Q)$(P∨Q) and $(\neg P \vee R)$(¬P∨R),

  cancel out P and ¬P → $Q \vee R$

This gives a new fact that must also be true.

Difference between : Resolution in propositional logic and Resolution in predicate logic

Feature	Propositional Logic	Predicate Logic
Works with	Simple propositions	Variables, quantifiers, predicates
Extra steps needed	None	CNF conversion, Skolemization, Unification
Power of reasoning	Limited (finite set of statements)	Much more powerful (general statements about objects)
Example	$(P \lor Q), (\neg P \lor R) \longrightarrow (Q \lor R)$	$\forall x\big(Man(x) \rightarrow Mortal(x)\big), Man(Socrates) \longrightarrow Mortal(Socrates)$

UPES
UNIVERSITY OF TOMORROW