Logical representation

1. Language,concrete rules, deals with propositions, no ambiguity.
2. Drawing a conclusion
3. Lays down some important communication rules.
4. Consist of precisely defined syntax and semantics (support inference)
5. Each knowledge can be translated into logic syntax and semantics.

Syntax	semantics
How can you construct sentences.	interprets
which symbol we can use in knowledge representation	meaning

Propositional logic

1. Declarative statement
2. Either true or false or both,
3. Logical and mathematical form

Types

Atomic Compound

Propositional logic is a branch of mathematics that studies the logical relationships between propositions (or statements, sentences, assertions) taken as a whole, and connected via logical connectives.

Example
1.it is Sunday,
2. The sun rises from west.
3. 3+3=7
4. 5 is a prime number.

4. Boolean logic.
5. Tautology (always true), contradiction (always false).
6. Questions, commands and opinions never be a propositional logic.

Some sentences that do not have a truth value or may have more than one truth value are not propositions. For Example,

1. What time is it?
2. Go out and Play
3. x + 1 = 2

Propositions constructed using one or more propositions are called compound propositions.

Logical connectives (operators).

1. Negations.
2. Conjunctions.
3. Disjunctions.
4. Implications.
5. Bidirectional.

Connective Symbols	Words	Terms	Example
∧	AND	Conjunction	A ∧ B
∨	OR	Disjunction	A ∨ B
→	Implies	Implication	A → B
⇔	If and only If	Biconditional	A ⇔ B
¬	Not	Negation	¬A

Truth table of propositional logic

Negation

P	Negation of p
T	F
F	T

Conjunction

P	q	P^q
T	T	T
T	F	F
F	T	F
F	F	F

Disjunction

P	q	P v a
T	T	T
T	F	T
F	T	T
F	F	F

Properties of operators

1. Commutativity.
2. Associativity.
3. Identity element.
4. Distributive.
5. De Morgan's law
6.double negation elimination.

Precedence of Connectives in Propositional Logic

When evaluating **compound propositions** with multiple logical connectives, it's important to follow a specific **order of precedence** to ensure accurate results. Similar to arithmetic operations, **logical operators** are evaluated in a defined sequence, from highest to lowest precedence.

Order of Precedence

1. **NOT (¬)** – Negation has the **highest precedence** and is evaluated first.
2. **AND (∧)** – Conjunction is evaluated next, after negations are resolved.
3. **OR (∨)** – Disjunction comes after AND operations.
4. **IF-THEN (→)** – Implication is evaluated after OR.
5. **IF AND ONLY IF (↔)** – Biconditional has the **lowest precedence**.

Example: Precedence in Action

Consider the following logical expression:

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

- **Step 1:** Evaluate **¬P** (Negation).
- **Step 2:** Evaluate **Q ∧ R** (AND).
- **Step 3:** Evaluate **¬P ∨ (Q ∧ R)** (OR).

The final result depends on the proper **evaluation order**, ensuring the correct outcome.

Logical Equivalence in Propositional Logic

1. **De Morgan's Laws:**
    - These laws show how **negations of conjunctions** and **disjunctions** behave:

$$\neg(P \wedge Q) \equiv (\neg P \vee \neg Q)$$

$$\neg(P \vee Q) \equiv (\neg P \wedge \neg Q)$$

2. **Double Negation:**
    - Negating a negation gives the original proposition:

$$\neg(\neg P) \equiv P$$

3. **Implication and Disjunction:**
    - An implication can be rewritten as:

$$P \rightarrow Q \equiv \neg P \vee Q$$

Tautologies and Contradictions

- **Tautology:**A tautology is a statement that is **always true**, no matter the truth values of its individual propositions.
  - **Example:** $P \lor \neg P \equiv \text{True}$
- **Contradiction:** A contradiction is a statement that is **always false**.
  - **Example:** $P \land \neg P \equiv \text{False}$

In propositional logic, **logical operators** follow specific properties that allow us to **manipulate and simplify logical expressions**. Understanding these properties is essential for building efficient AI systems that rely on logical reasoning.

De Morgan's Laws

These laws describe how **negations** distribute over **AND (∧)** and **OR (∨)** operations:

- **First Law:**
  - $\neg(P \land Q) \equiv (\neg P \lor \neg Q)$

This means that the negation of a conjunction is equivalent to the disjunction of the negated propositions.

- **Second Law:**
  - $\neg(P \lor Q) \equiv (\neg P \land \neg Q)$

This means that the negation of a disjunction is equivalent to the conjunction of the negated propositions.

2. Commutative Property

This property states that the **order of the propositions** does not affect the result of **AND (∧)** and **OR (∨)** operations:

- **AND:**
  - $P \land Q \equiv Q \land P$
- **OR:**
  - $P \lor Q \equiv Q \lor P$

3. Associative Property

This property allows us to **group propositions** in any order when using **AND** or **OR** operations:

- **AND:**
  - $(P \land Q) \land R \equiv P \land (Q \land R)$

- **OR:**
  - $(P \lor Q) \lor R \equiv P \lor (Q \lor R)$

4. Distributive Property

This property states that **AND distributes over OR**, and vice versa:

- **AND over OR:**
  - $P \land (Q \lor R) \equiv (P \land Q) \lor (P \land R) P$
- **OR over AND:**
  - $P \lor (Q \land R) \equiv (P \lor Q) \land (P \lor R)$

Applications of Propositional Logic in AI

1. Knowledge Representation in Expert Systems:

- Represents **rules and facts** to solve domain-specific problems (e.g., medical diagnosis systems).

2. Reasoning and Decision-Making:

- AI agents use logical rules to make **decisions** (e.g., robot vacuum cleaners deciding when to start cleaning).

3. Natural Language Processing (NLP):

- Helps analyze text and respond logically (e.g., chatbots understanding weather-related queries).

4. Game-Playing AI:

- Uses logic to **make strategic moves** (e.g., deciding checkmate in chess).

Limitations of Propositional Logic

1. Inability to Handle Complex Relationships

- Propositional logic cannot represent **relationships between multiple objects** or deal with hierarchies of information.

2. No Handling of Uncertainty

- It works only with **true or false** values and cannot deal with **probabilities or uncertain outcomes**, limiting its use in real-world applications involving incomplete data.

3. Limited Expressiveness

- It cannot represent **time-based sequences** or dynamic events, which are crucial in some AI systems like speech recognition and robotics.

4. Scalability Issues

- As the number of propositions grows, the **complexity of expressions** increases, making reasoning slower and harder to manage.

https://www.appliedaicourse.com/blog/propositional-logic-in-artificial-intelligence/