International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** I **Month of publication:** January 2026

DOI: https://doi.org/10.22214/ijraset.2026.77026

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue I Jan 2026- Available at www.ijraset.com

Knowledge Base Agents in Artificial Intelligence

Nethravathi H [1], Mangesh S Pai[2], Raziya Sultana[3], Siddarth Y K[4], Tejaswini B H[5], Manoj H M[6], Kirankumar M G[7], Divya R[8]
[1, 3, 4, 5, 6, 7, 8]Department of Computer Science, D V S College of Arts and Science, Shimoga, Karnataka, India
[2]Department of Computer Science, [2]Kateel Ashok Pai Memorial College, Shimoga, Karnataka, India

Abstract: Knowledge Representation (KR) in AI focuses on structuring real-world information (facts, concepts, rules) into formal systems (logic, graphs, ontologies) that computers can process for tasks like reasoning, planning, and problem-solving, bridging the gap between human understanding and machine computation by making knowledge explicit and usable for intelligent agents.
Keywords: Knowledge, Intelligent agents, Representation, Reasoning, Human and Machine.

I. INTRODUCTION

Knowledge Representation (KR) in Artificial Intelligence (AI) is the crucial process of encoding real-world information, facts, and rules into computer-readable formats, allowing AI systems to understand, reason, make decisions, and solve complex problems like humans do, going beyond simple data storage to enable true intelligence. It bridges the gap between raw data and meaningful action by structuring knowledge, enabling inference, and facilitating tasks in areas like healthcare, robotics, and language understanding.

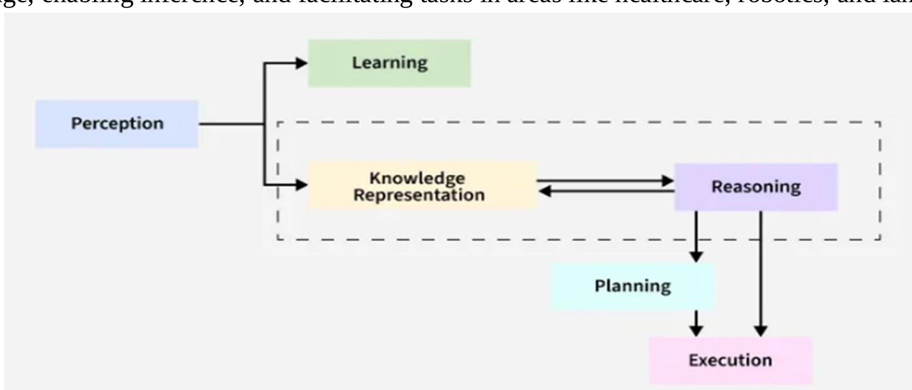


Fig 1: Knowledge Representation in AI

II. KNOWLEDGE-BASED AGENTS

The central component of a knowledge-based agent is its knowledge base, or KB. Informally, a knowledge base is a set of sentences. (Here "sentence" is used as a technical term.) Each sentence is expressed in a language called a knowledge representation language and represents some assertion about the world.

There must be a way to add new sentences to the knowledge base and a way to query what is known. The standard names for these tasks are TELL and ASK, respectively. Both tasks may involve inference-that is, deriving new sentences from old.

Figure below shows the outline of a knowledge-based agent program.

```
function KB-AGENT(percept) returns an action
   static: KB, a knowledge base
           t, a counter, initially 0, indicating time

   TELL(KB, MAKE-PERCEPT-SENTENCE(percept, t))
   action ← ASK(KB, MAKE-ACTION- QUERY(^))
   TELL(KB, MAKE-ACTION-SENTENCE(action, t))
   t ← t + 1
   return action
```

Fig 2 : Agent Program

Like all our agents, it takes a percept as input and returns an action. The agent maintains a knowledge base, KB, which may initially contain some background knowledge. Each time the agent program is called, it does three things.

1) First, it TELLS the knowledge base what it perceives.
2) Second, it ASKS the knowledge base what action it should perform. In the process of answering this query, extensive reasoning may be done about the current state of the world, about the outcomes of possible action sequences, and so on.
3) Third, the agent records its choice with TELL and executes the action. The second TELL is necessary to let the knowledge base know that the hypothetical action has actually been executed.

The details of the representation language are hidden inside three functions that implement the interface between the sensors and actuators and the core representation and reasoning system.

a) MAKE-PERCEPT-SENTENCE constructs a sentence asserting that the agent perceived the given percept at the given time.
b) MAKE-ACTION-QUERY constructs a sentence that asks what action should be done at the current time. Finally,
c) MAKE-ACTION-SENTENCE constructs a sentence asserting that the chosen action was executed.

The agent's initial program, before it starts to receive percepts, is built by adding one by one the sentences that represent the designer's knowledge of the environment.

- Designing the representation language to make it easy to express this knowledge in the form of sentences simplifies the construction problem enormously. This is called the declarative approach to system building.
- In contrast, the procedural approach encodes desired behaviours directly as program code; minimizing the role of explicit representation and reasoning can result in a much more efficient system.

A. *The Wumpus World*

The Wumpus world is a cave consisting of rooms connected by passageways. Lurking somewhere in the cave is the wumpus, a beast that eats anyone who enters its room. The wumpus can be shot by an agent, but the agent has only one arrow. Some rooms contain bottomless pits that will trap anyone who wanders into these rooms (except for the wumpus, which is too big to fall in). The only mitigating feature of living in this environment is the possibility of finding a heap of gold.
A sample wumpus world is shown in Figure below.

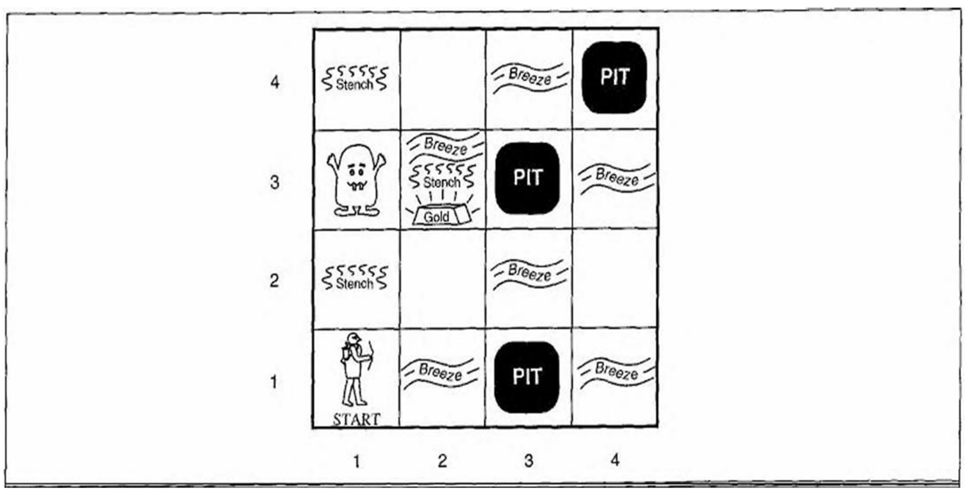


Fig 3 : Wumpus World Representation

The definition of the task environment is given, by the PEAS description:

1) Performance Measure: +I000 for picking up the gold, -1000 for falling into a pit or being eaten by the wumpus, -1 for each action taken and -10 for using up the arrow.
2) Environment: A 4 x 4 grid of rooms. The agent always starts in the square labeled [1,1], facing to the right. The locations of the gold and the wumpus are chosen randomly, with a uniform distribution, from the squares other than the start square. In addition, each square other than the start can be a pit.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue I Jan 2026- Available at www.ijraset.com

3) Actuators: The agent can move forward, turn left by 90°, or turn right by 90°. The agent dies a miserable death if it enters a square containing a pit or a live wumpus. Moving forward has no effect if there is a wall in front of the agent.

- The action Grab can be used to pick up an object that is in the same square as the agent
- The action Shoot can be used to fire an arrow in a straight line in the direction the agent is facing. The arrow continues until it either hits (hence kills) the wumpus or hits a wall. The agent only has one arrow, so only the first Shoot action has any effect.

4) Sensors

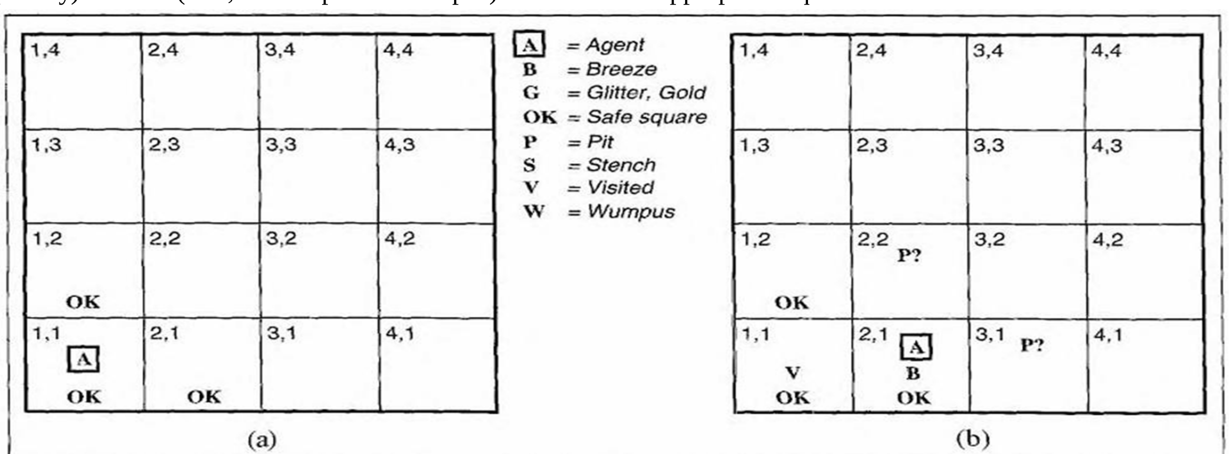
The agent has five sensors, each of which gives a single bit of information:

1) In the square containing the wumpus and in the directly (not diagonally) adjacent squares the agent will perceive a stench.
2) In the squares directly adjacent to a pit, the agent will perceive a breeze.
3) In the square where the gold is, the agent will perceive a glitter.
4) When an agent walks into a wall, it will perceive a bump.
5) When the wumpus is killed, it emits a woeful scream that can be perceived anywhere in the cave.

The percepts will be given to the agent in the form of a list of five symbols; for example, if there is a stench and a breeze, but no glitter, bump, or scream, the agent will receive the percept [Stench, Breeze, None, None, None].

Let us watch a knowledge-based wumpus agent exploring the environment shown in Figure above. The agent's initial knowledge base contains the rules of the environment, as listed previously; in particular, it knows that it is in [1,1] and that [1,1] is a safe square.

The first percept is [None, None, None, None, None], from which the agent can conclude that its neighboring squares are safe. Figure (a) shows the agent's state of knowledge at this point. We list (some of) the sentences in the knowledge base using letters such as B (breezy) and OK (safe, neither pit nor wumpus) marked in the appropriate squares.



The first step taken by the agent in the wumpus world. (a) The initial situation, after percept *[None,None, None, None, None]*. (b) After one move, with percept *[None,Breeze, None, None, None]*.

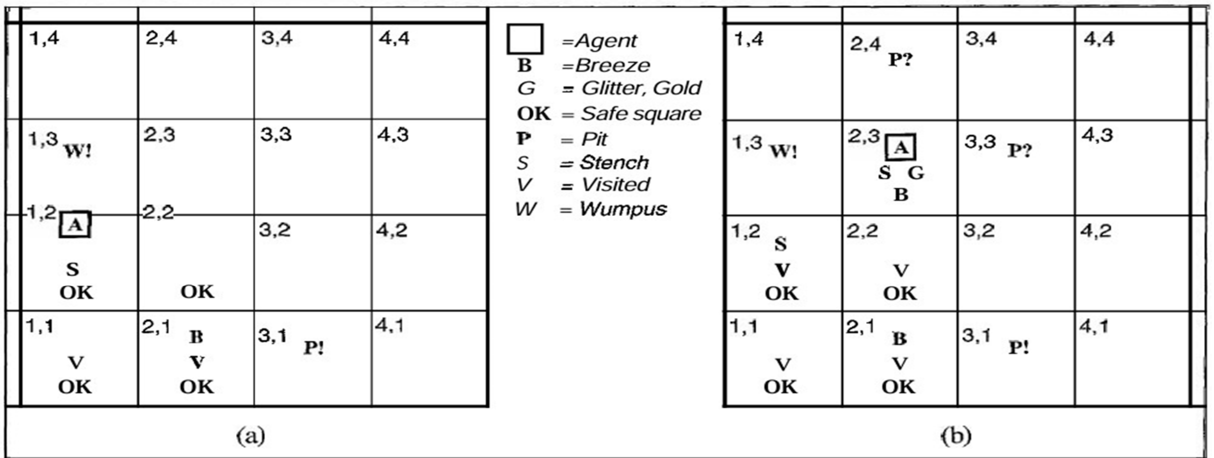
Fig 3.1 : Wumpus World Representation

From the fact that there was no stench or breeze in [1,1], the agent can infer that [1,2] and [2,1] are free of dangers. They are marked with an OK to indicate this. A cautious agent will move only into a square that it knows is OK. Let us suppose the agent decides to move forward to [2,1] , giving the scene in Figure (b).

The agent detects a breeze in [2, 1], so there must be a pit in a neighboring square. The pit cannot be in [1, 1], by the rules of the game, so there must be a pit in [2,2] or [3,1] or both. The notation P? In Figure (b) indicates a possible pit in those squares. At this point, there is only one known square that is OK and has not been visited yet. So the agent will turn around, go back to [1, 1], and then proceed to [1,2].

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue I Jan 2026- Available at www.ijraset.com

The new percept in [1,2] is [Stench, None, None, None, None], resulting in the state of knowledge shown in below Figure (a).



Two later stages in the progress of the agent. (a) After the third move, with percept *[Stench,None, None, None, None]*. (b) After the fifth move, with percept *[Stench,Breeze, Glitter, None, None]*.

Fig 3.2 : Wumpus World Representation

The stench in [1,2] means that there must be a wumpus nearby. But the wumpus cannot be in [1,1], by the rules of the game, and it cannot be in [2,2] (or the agent would have detected a stench when it was in [2,1]). Therefore, the agent can infer that the wumpus is in [1,3]. The notation W! Indicates this. Moreover, the lack of a Breeze in [1,2] implies that there is no pit in [2,2]. Yet we already inferred that there must be a pit in either [2,2] or [3,1], so this means it must be in [3,1].
The agent has now proved to itself that there is neither a pit nor a wumpus in [2,2], so it is OK to move there. We will not show the agent's state of knowledge at [2,2]; we just assume that the agent turns and moves to [2,3], giving us Figure (b). In [2,3], the agent detects a glitter, so it should grab the gold and thereby end the game.
In each case where the agent draws a conclusion from the available information, that conclusion is guaranteed to be correct the available information is correct. This is a fundamental property of logical reasoning.

B. Logic

A logic must also define the semantics of the language. Loosely speaking, semantics has to do with the "meaning" of sentences. In logic, the definition is more precise. The semantics of the language defines the truth of each sentence with respect to each possible world. For example, the usual semantics adopted for arithmetic specifies that the sentence "x + y = 4" is true in a world where x is 2 and y is 2, but false in a world where x is 1 and y is 1.' In standard logics, every sentence must be either true or false in each possible world-there is no "in between." When we need to be precise, we will use the term mode in place of "possible world." Models are mathematical abstractions, each of which simply fixes the truth or falsehood of every relevant sentence. For example, and the sentence x + y = 4 is true when there are four in total
Now that we have a notion of truth, we are ready to talk about logical reasoning. This involves the relation of logical entailment between sentences - the idea that a sentence follows 1ogically from another sentence. In mathematical notation, we write as

$$\alpha \models \beta$$

The formal definition of entailment is α |= β if and only if, in every model in which α is true, β is also true. Another way to say this is that if α is true, then β must also be true. Informally, the truth of β is "contained" in the truth of α.

C. Propositional Logic

Propositional logic (PL) is the simplest form of logic where all the statements are made by propositions. A proposition is a declarative statement which is either true or false. It is a technique of knowledge representation in logical and mathematical form.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue I Jan 2026- Available at www.ijraset.com

1) Syntax

The syntax of propositional logic defines the allowable sentences. The atomic sentences the indivisible syntactic elements-consist of a single proposition symbol. Each such symbol stands for a proposition that can be true or false. We will use uppercase names for symbols: P, Q, R, and so on. For example, we might use $W_{1,3}$ to stand for the proposition that the wumpus is in [1, 3]. There are two proposition symbols with fixed meanings:

- True is the always-true proposition and
- False is the always-false proposition.

Complex sentences are constructed from simpler sentences using logical connectives. There are five connectives in common use:

a) ¬ (not) Negation: A sentence such as ¬ P is called negation of P. A literal can be either Positive literal or negative literal.

b) A (and) Conjunction: A sentence which has ∧ connective such as, P ∧ Q is called a Conjunction.
Example: Rohan is intelligent and hardworking. It can be written as P= Rohan is intelligent Q= Rohan is hardworking, then P∧ Q.

c) V (or) Disjunction: A sentence which has ∨ connective, such as P ∨ Q is called disjunction.
Example: "Ritika is a doctor or Engineer", Here P= Ritika is Doctor. Q= Ritika is Doctor, so we can write it as P ∨ Q.

d) ==> (implies) Implication: A sentence such as P → Q, is called an implication. Implications are also known as if-then rules.
It can be represented as, if it is raining, then the street is wet. Let P= It is raining, and Q= Street is wet, so it is represented as P → Q ⇔ (if and only if) Biconditional: A sentence such as P⇔ Q is a Biconditional sentence, for
Example, If I am breathing, then I am alive. P= I am breathing, Q= I am alive, it can be represented as P ⇔ Q.

2) Semantics

The semantics defines the rules for determining the truth of a sentence with respect to a particular model. In propositional logic, a model simply fixes the truth value - true or false - for every proposition symbol. For example, if the sentences in the knowledge base make use of the proposition symbols $P_{1,2}$ $P_{2,2}$, and $P_{3,1}$, then one possible model is

$$m_1 = \{P_{1,2}=\text{false},\ P_{2,2}=\text{false},\ P_{3,1}=\text{true}) .$$

The semantics for propositional logic must specify how to compute the truth value of any sentence, given a model. We need to specify how to compute the truth of atomic sentences and how to compute the truth of sentences formed with each of the five connectives. Atomic sentences are easy:

- True is true in every model and false is false in every model.
- The truth value of every other proposition symbol must be specified directly in the model. For example, in the model m_1 given earlier, $P_{1,2}$ is false.

For complex sentences, we have rules such as

- For any sentence s and any model m, the sentence is ~s true in m if and only if s is false in m.

The rules for each connective can be summarized in a truth table that specifies the truth value of a complex sentence for each possible assignment of truth values to its components. Truth tables for the five logical connectives are given in Figure below.

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*

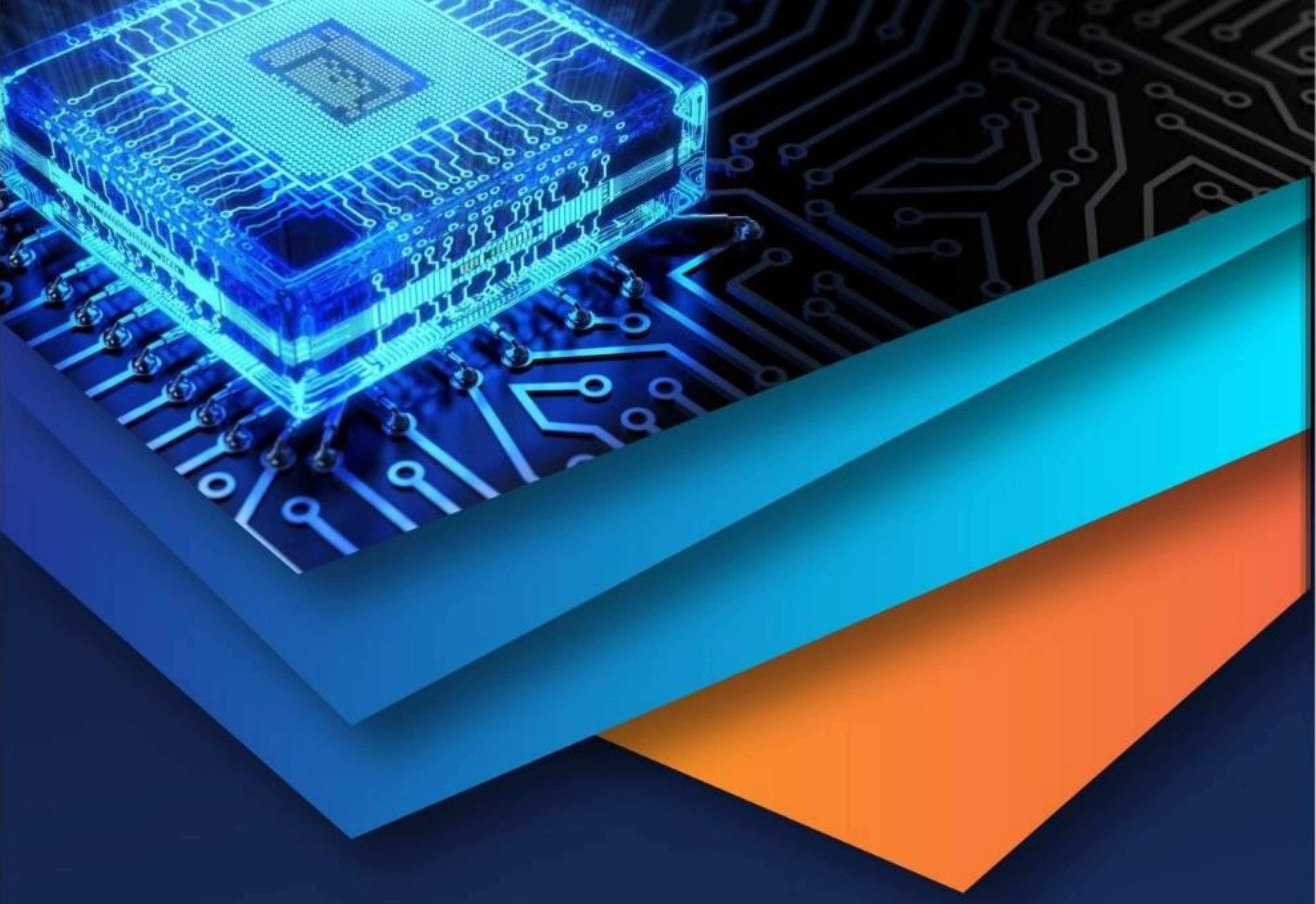
Table 1 : Connectives

III. CONCLUSION

Knowledge-based agents in AI are a major breakthrough. They can reason, learn, and make decisions, changing how we use machines. These agents are more than just tools—they inspire innovation. From healthcare to finance, they create smarter solutions. Of course, challenges like ethics and scalability exist. But the benefits far outweigh the problems. As research advances, combining human skills with AI will unlock amazing possibilities. Whether it's simplifying tasks or solving tough problems, these agents are ready to change the game. They aren't just improving technology—they're reshaping how we live, work, and grow.

REFERENCES

[1] Artificial Intelligence – A modern approach : Stuart Russell, Peter Norvig, 2nd Edition
[2] https://www.geeksforgeeks.org/artificial-intelligence/knowledge-representation-in-ai/
[3] https://www.youngurbanproject.com/knowledge-based-agents-in-ai/



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY