Artificial Intelligence

Unit IV

Knowledge Representation

- Intelligent agent acquire knowledge from the environment and further build the knowledge with reference to the problem.
- It is this knowledge which is exploited further with the actions and decisions.
- Thus, appropriate and precise representation of the knowledge becomes a critical factor in the process.
- What is knowledge? It is some set of patterns or associations derived from the data or information that helps in making decisions and resolves problems.
- A systematic reasoning process is required when we try to relate the events to the outcomes or to arrive judgement.

Approaches of Knowledge Representation

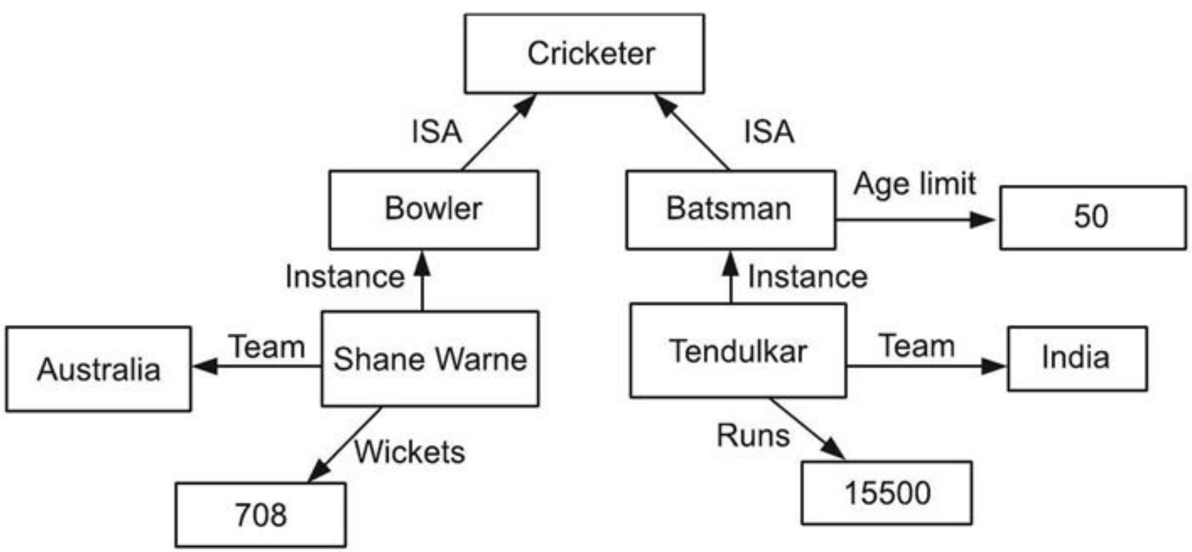
1.Simple relational knowledge:

- It is the simplest way of storing facts which uses the relational method, and each fact about a set of the object is set out systematically in columns.
- This approach of knowledge representation is famous in database systems where the relationship between different entities is represented.

Employee	*Salary*	*Experience*
Sameer	30000	3
Kavita	20000	2
Jasmin	20000	2

2. Inheritable knowledge

- In the inheritable knowledge approach, all data must be stored into a hierarchy of classes.
- This approach use inheritance property.
- Elements inherit values from other members of a class.
- This approach contains inheritable knowledge which shows a relation between instance and class, and it is called instance relation.



3. Inferential Knowledge

- Inferential knowledge refers to the ability to draw logical conclusions or make predictions based on available data or information. Predicate logic is used.

  Example: Every student smiles

  $\forall x\ (student(x) \rightarrow smile(x))$

4. Procedural Knowledge

- Procedural knowledge approach uses small programs and codes which describes how to do specific things, and how to proceed.
- In this approach, one important rule is used which is **If-Then rule.**

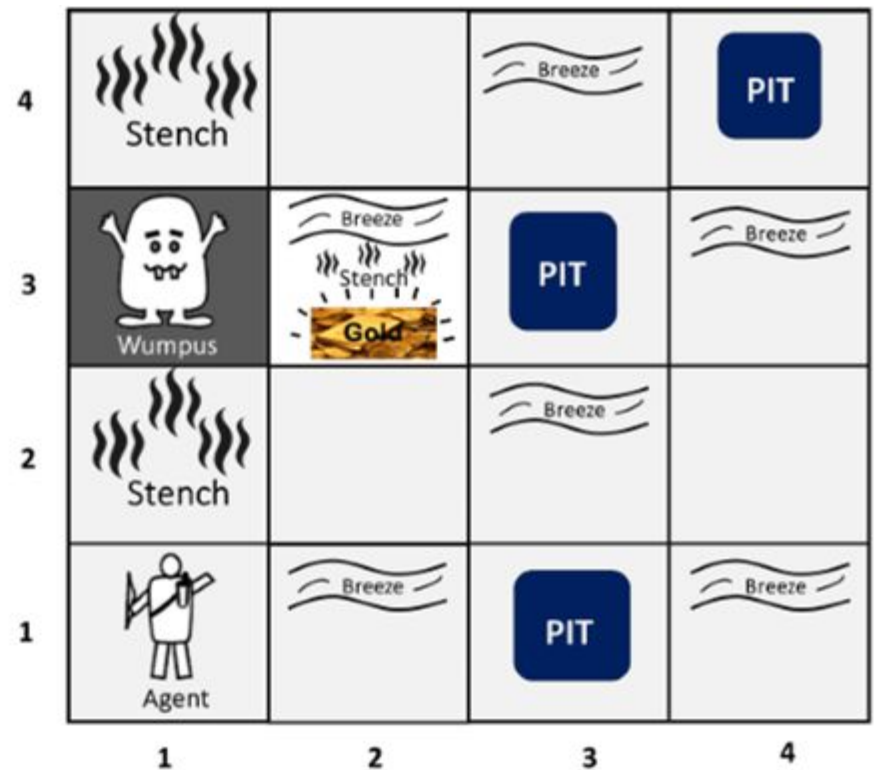
Issues in knowledge representation

- Attributes are the most important ones that have an impact on the representation.
- It is required to understand and identify the important attributes.
- Identifying the relationships among the attributes in the representation is equally important.
- Level or upto what depth the mapping of the knowledge is to be doe defines the granularity.

The Wumpus World

- The Wumpus world is a simple world example to illustrate the worth of a knowledge-based agent and to represent knowledge representation.
- The Wumpus world is a cave which has 4/4 rooms connected with passageways. So there are total 16 rooms which are connected with each other.
- We have a knowledge-based agent who will go forward in this world. The cave has a room with a beast which is called Wumpus, who eats anyone who enters the room.
- The Wumpus can be shot by the agent, but the agent has a single arrow. In the Wumpus world, there are some Pits rooms which are bottomless, and if agent falls in Pits, then he will be stuck there forever.

- The exciting thing with this cave is that in one room there is a possibility of finding a heap of gold. So the agent goal is to find the gold and climb out the cave without fallen into Pits or eaten by Wumpus.
- The agent will get a reward if he comes out with gold, and he will get a penalty if eaten by Wumpus or falls in the pit.



There are also some components which can help the agent to navigate the cave. These components are given as follows:

- The rooms adjacent to the Wumpus room are smelly, so that it would have some stench.
- The room adjacent to PITs has a breeze, so if the agent reaches near to PIT, then he will perceive the breeze.
- There will be glitter in the room if and only if the room has gold.
- The Wumpus can be killed by the agent if the agent is facing to it, and Wumpus will emit a horrible scream which can be heard anywhere in the cave.

PEAS description of Wumpus world

Performance measure:

- +1000 reward points if the agent comes out of the cave with the gold.
- -1000 points penalty for being eaten by the Wumpus or falling into the pit.
- -1 for each action, and -10 for using an arrow.
- The game ends if either agent dies or came out of the cave.

Environment:

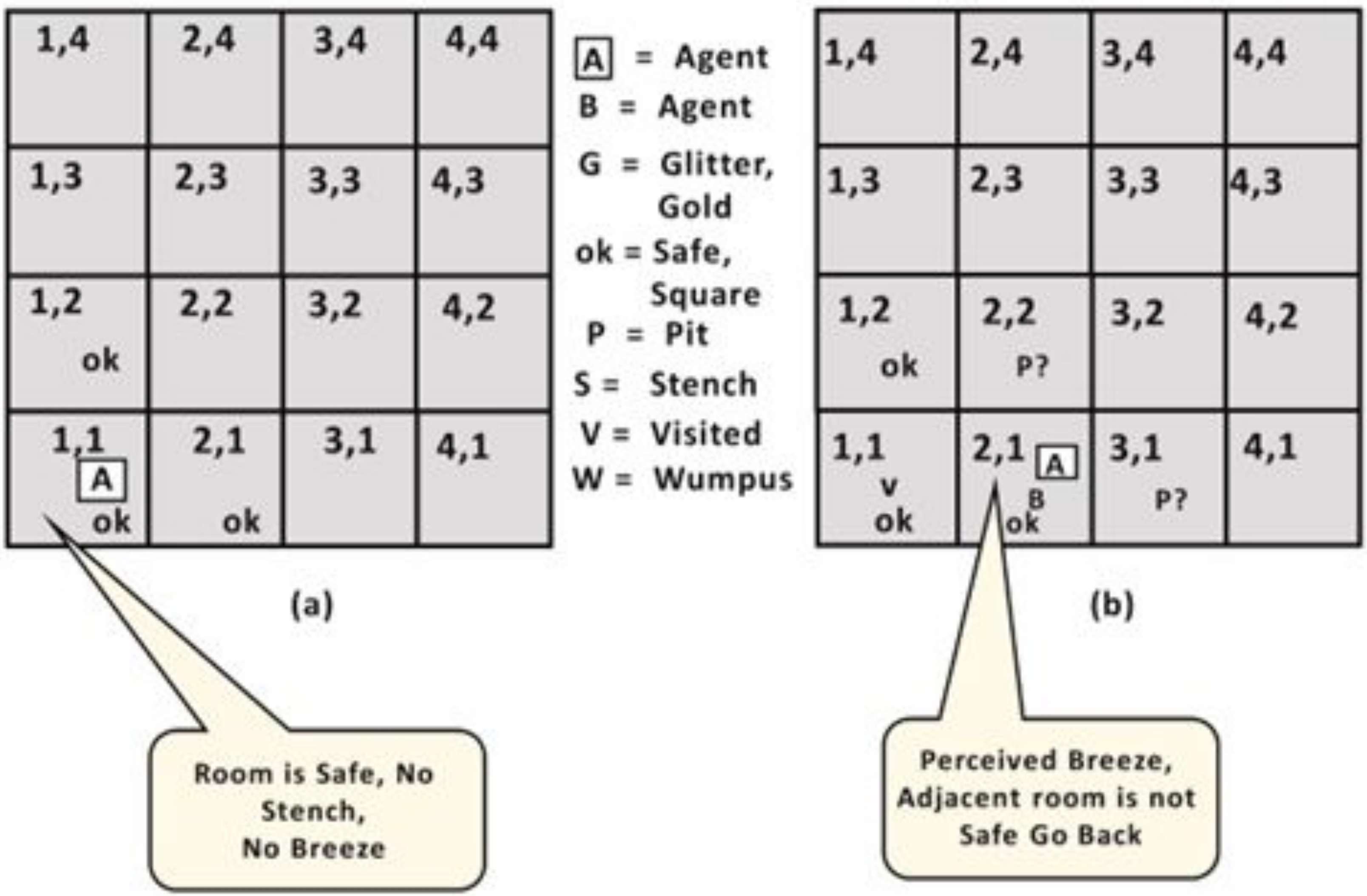
- A 4*4 grid of rooms.
- The agent initially in room square [1, 1], facing toward the right.
- Location of Wumpus and gold are chosen randomly except the first square [1,1].

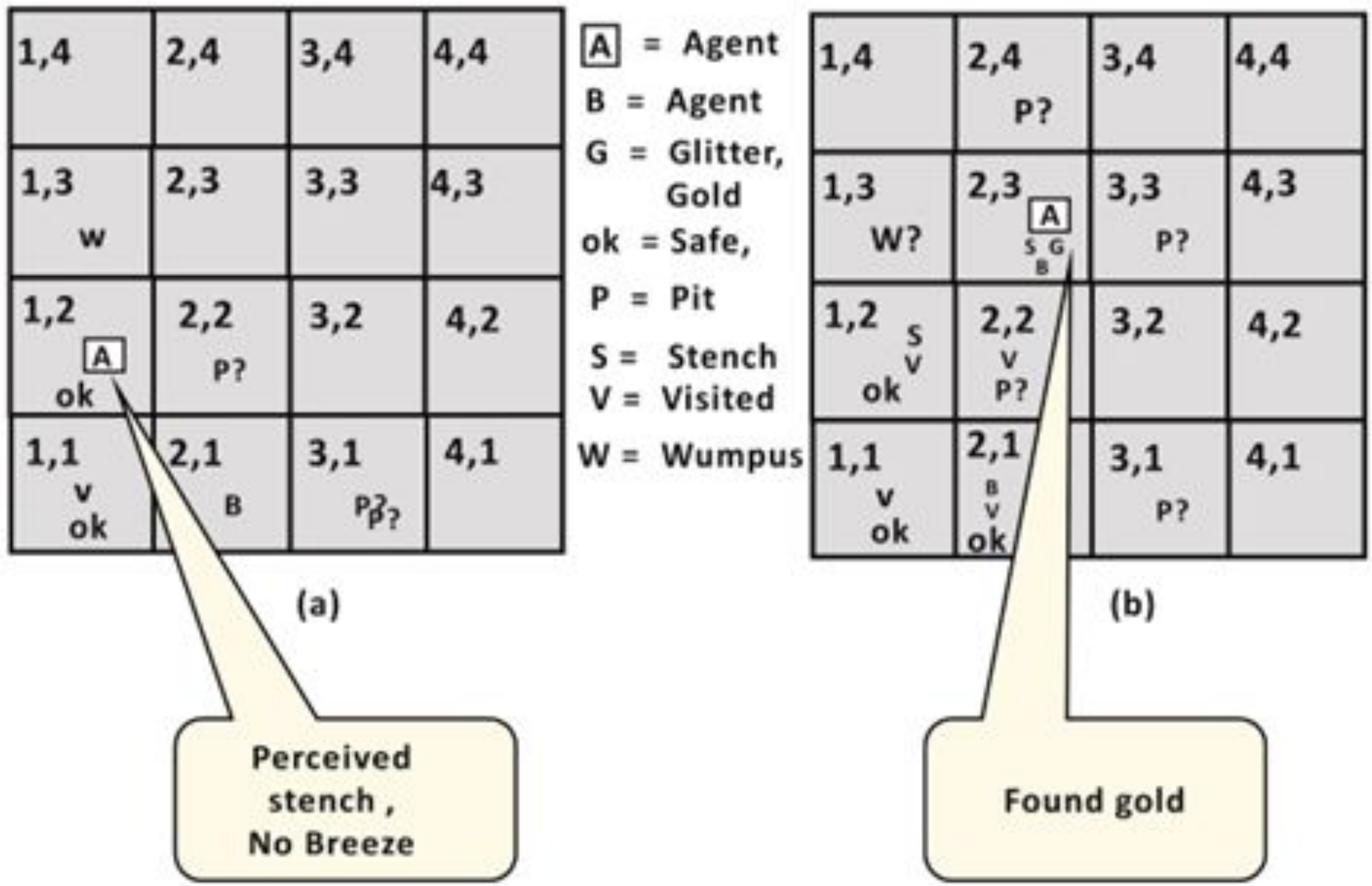
Actuators:

- Left turn,
- Right turn
- Move forward
- Grab
- Release
- Shoot.

Sensors:

- The agent will perceive the **stench** if he is in the room adjacent to the Wumpus. (Not diagonally).
- The agent will perceive **breeze** if he is in the room directly adjacent to the Pit.
- The agent will perceive the **glitter** in the room where the gold is present.
- The agent will perceive the **bump** if he walks into a wall.
- When the Wumpus is shot, it emits a horrible **scream** which can be perceived anywhere in the cave.


1,4
2,4
3,4
4,4
1,3
2,3
3,3
4,3
1,2
ok
2,2
3,2
4,2
1,1
A
ok
2,1
ok
3,1
4,1
(a)
Room is Safe, No Stench, No Breeze
A = Agent
B = Agent
G = Glitter, Gold
ok = Safe, Square
P = Pit
S = Stench
V = Visited
W = Wumpus
1,4
2,4
3,4
4,4
1,3
2,3
3,3
4,3
1,2
ok
2,2
P?
3,2
4,2
1,1
V
ok
2,1
A
B
ok
3,1
P?
4,1
(b)
Perceived Breeze, Adjacent room is not Safe Go Back


1,4
2,4
3,4
4,4
1,3
W
2,3
3,3
4,3
1,2
A
ok
2,2
P?
3,2
4,2
1,1
V
ok
2,1
B
3,1
P?
4,1
(a)
Perceived stench , No Breeze
A = Agent
B = Agent
G = Glitter, Gold
ok = Safe,
P = Pit
S = Stench
V = Visited
W = Wumpus
1,4
2,4
P?
3,4
4,4
1,3
W?
2,3
A
S G B
3,3
P?
4,3
1,2
S V
ok
2,2
V
P?
3,2
4,2
1,1
V
ok
2,1
B V
ok
3,1
P?
4,1
(b)
Found gold

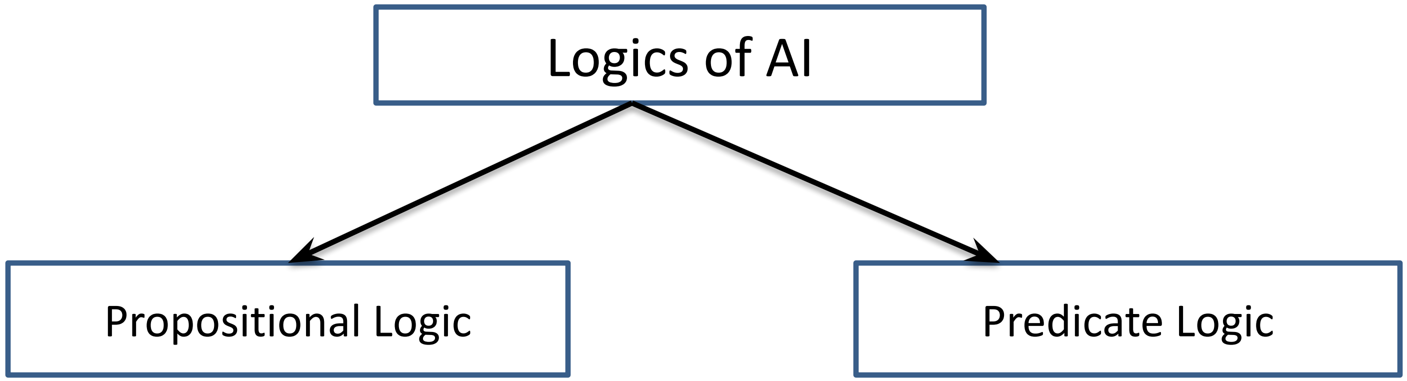
Knowledge base for Wumpus world

- Let $\mathbf{P}_{i,j}$ be true if there is a Pit in the room [i, j].
- Let $\mathbf{B}_{i,j}$ be true if agent perceives breeze in [i, j], (dead or alive).
- Let $\mathbf{W}_{i,j}$ be true if there is wumpus in the square[i, j].
- Let $\mathbf{S}_{i,j}$ be true if agent perceives stench in the square [i, j].
- Let $\mathbf{V}_{i,j}$ be true if that square[i, j] is visited.
- Let $\mathbf{G}_{i,j}$ be true if there is gold (and glitter) in the square [i, j].
- Let $\mathbf{OK}_{i,j}$ be true if the room is safe.

Representation of Knowledgebase for Wumpus world:

Following is the Simple KB for wumpus world when an agent moves from room [1, 1], to room [2,1]:

$\neg W_{11}$	$\neg S_{11}$	$\neg P_{11}$	$\neg B_{11}$	$\neg G_{11}$	V_{11}	OK_{11}
$\neg W_{12}$	----	$\neg P_{12}$	-----	----	$\neg V_{12}$	OK_{12}
$\neg W_{21}$	$\neg S_{21}$	$\neg P_{21}$	B_{21}	$\neg G_{21}$	V_{21}	OK_{21}



- When and where these logics are used?
- Our main aim is to create an artificially intelligent system. We can say that our machine is artificially intelligent, if it has the capability to reason (logic).
- How a machine will have the reasoning capability? only if it has the proper input.
- The input in AI machine will be a knowledge base, from that the machine will learn and able to think or reason and produce us the proper output.

- The knowledge that is provided to the machine will be in natural language (English).
- Machines cannot able to understand the natural language, hence the knowledge has to be converted to logical representation, for the machines to understand easily.
- Hence we are going for propositional and predicate logics.
- Example: Spot is a dog

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.

Example:

- The Sun rises from West (False proposition)
- The Sun rises from East(True proposition)
- 3+3= 7(False proposition)
- 5 is a prime number.   (True proposition)

<u>Following are some basic facts about propositional logic:</u>

- Propositional logic is also called Boolean logic as it works on 0 and 1.
- In propositional logic, we use symbolic variables to represent the logic, and we can use any symbol for a representing a proposition, such A, B, C, P, Q, R, etc.
- Propositions can be either true or false, but it cannot be both.
- The propositions and connectives are the basic elements of the propositional logic.
- A proposition formula which is always true is called **tautology**, and it is also called a valid sentence.
- A proposition formula which is always false is called **Contradiction**.
- A proposition formula which has at least one true is called **satisfiability**.
- Statements which are questions, commands, or opinions are not propositions such "**How are you?**", "**What is your name?**", are not propositions.

Syntax of propositional logic

- The syntax of propositional logic defines the allowable sentences for the knowledge representation. There are two types of Propositions:
  - **Atomic Propositions**
  - **Compound propositions**

Atomic Proposition: Atomic propositions are the simple propositions. It consists of a single proposition symbol. These are the sentences which must be either true or false.

Example:

a) 2+2 is 4, it is an atomic proposition as it is a **true** fact.

b) "The Sun is cold" is also a proposition as it is a **false** fact.

Compound proposition: Compound propositions are constructed by combining simpler or atomic propositions, using parenthesis and logical connectives.

a) "It is raining today, and street is wet."

b) "Ankit is a doctor, and his clinic is in Mumbai."

Logical Connectives

- **Negation:** A sentence such as ¬ P is called negation of P
- **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. → P∧ Q.
- **Disjunction:** A sentence which has ∨ connective, such as **P ∨ Q**. is called disjunction, where P and Q are the propositions.
- **Example: "Ritika is a doctor or Engineer"**,
  Here P= Ritika is Doctor. Q= Ritika is Engineer, so we can write it as **P ∨ Q**.
- **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
- **Bi conditional:** A sentence such as **P⇔ Q is a Bi conditional sentence, example If I am breathing, then I am alive**
  P= I am breathing, Q= I am alive, it can be represented as P ⇔ Q.

Truth Table

For Negation:

P	¬ P
True	**False**
False	**True**

For Conjunction:

P	Q	P∧Q
True	True	**True**
True	False	**False**
False	True	**False**
False	False	**False**

For disjunction:

P	Q	P V Q.
True	True	**True**
False	True	**True**
True	False	**True**
False	False	**False**

For Implication:

P	Q	P→ Q
True	True	**True**
True	False	**False**
False	True	**True**
False	False	**True**

For Biconditional:

P	Q	$P \Leftrightarrow Q$
True	True	**True**
True	False	**False**
False	True	**False**
False	False	**True**

Predicate Logic

- Propositional logic can only be used to represent true or false statement and it is very difficult to represent complex statements.
- Predicate logic is also called as First order logic (FOL). It is an extension of propositional logic.
- FOL includes: object, relations, function, variables
- Object: people, numbers, colors, A,B,C
- Variables: x, y, z, a,b,c…
- Relation: it tells about the relationship between objects
- Function: example- friends(john, bill) – this kind of representation is called fact representation.
- Connectives: $\wedge$, $\vee$, $\neg$, $\equiv$, $\rightarrow$
- Quantifiers: $\forall$, $\exists$ {$\forall$ - universal quantifier, $\exists$- existential quantifier}
- FOL or FOPL includes syntax and semantics like a natural language.

- When we want to represent atomic sentence in FOL, it will follow some syntax
  - Syntax: Predicate(term1, term2…,term n)

    Function name   Constant or object

  - Example: “Bill and John are friends” can be represented as

    friends(bill, john)

    “ Dove is a bird” can be represented as

    bird(dove)

- If we want to represent sentences with “Some” “All” then we go for existential and universal quantifiers respectivley.
  - Example: “All students are intelligent” “Some students are naughty”

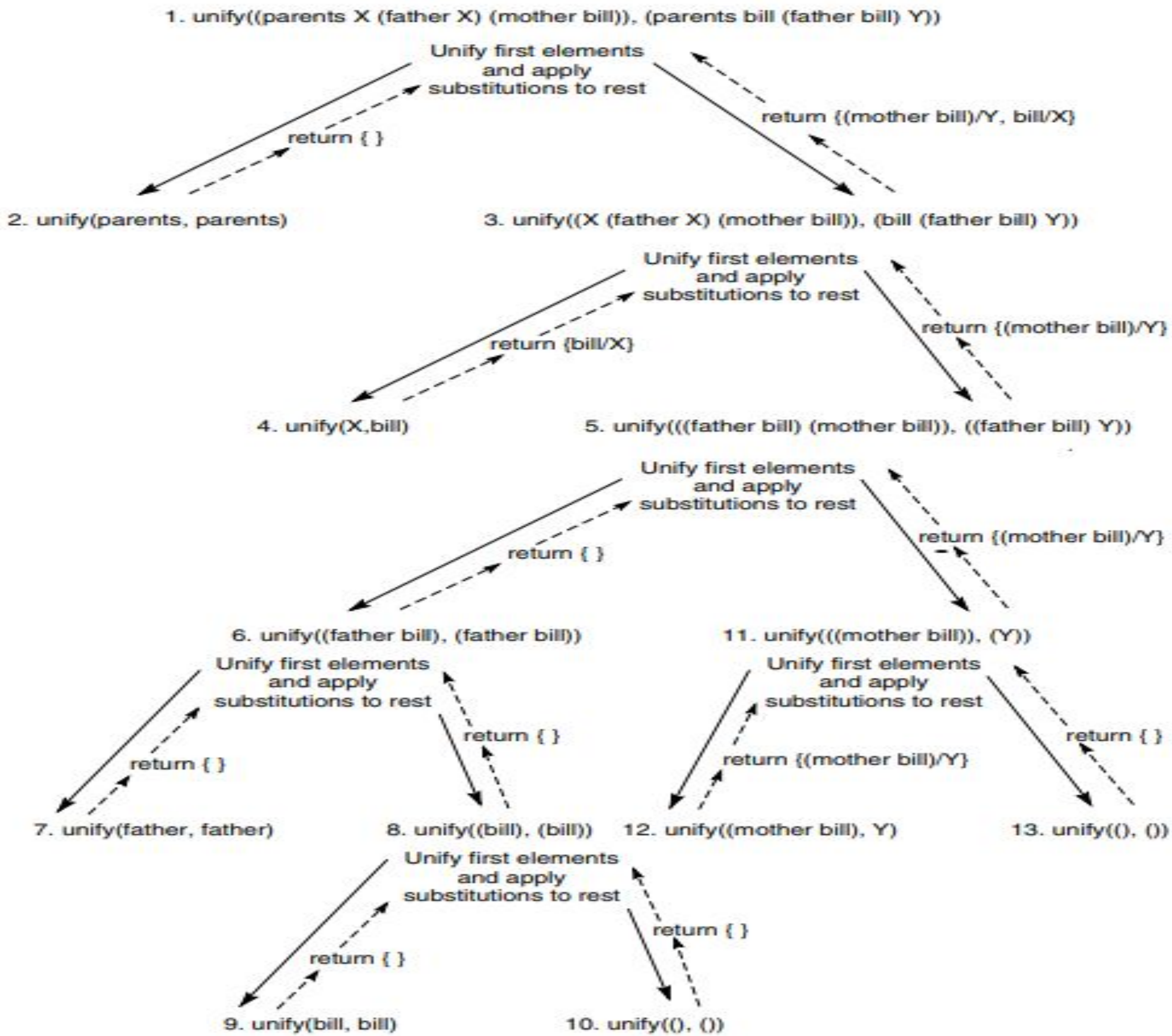
Examples

1. Mary loves everyone- $\forall x$ love (Mary, x)
2. No one talks- $\forall x \neg$talk(x)
3. Everyone loves himself- $\forall x$ love (x, x)
4. Everyone loves everyone- $\forall x \forall y$ love (x, y)
5. Every student smiles- $\forall x$ (student(x) $\rightarrow$ smile(x))
6. Every student except George smiles- $\forall x$ ((student(x) & x $\neq$ George) $\rightarrow$ smile(x))
7. Everyone walks or talks- $\forall x$ (walk (x) $\vee$ talk (x))
8. Every student who loves Mary is happy- $\forall x$ ((student(x) & love (x, Mary)) $\rightarrow$ happy (x)))

Unification in First Order Logic

- Unification is the process of making two different logical atomic expressions identical by finding a substitution.
- Unification depends on the substitution process.
- Let α_1 and α_2 are the two atomic sentences and Θ be a unifier such that $\alpha_1 \Theta = \alpha_2 \Theta$.
- Example: Find the MGU (Most General Unifier) for UNIFY{King(x), King(John)}.
- Substitution Θ = {John/x} is a unifier for these atoms and applying this substitution, both the expression will be identical.
- The UNIFY algorithm is used for unification, which takes two atomic sentences as input and returns an unifier for those sentences (if any exist).
- It returns “fails” if the expression do not match with each other.
- The substitution variables are called Most General Unifier or MGU.

- Lets consider another example, P(x,y) and P(a,f(z)), in this example we need to make both above statements identical to each other.
  - P(x,y) …..(i)
  - P(a,f(z))….(ii)
- Substitute x with a and y with f(z) and it will be represented as a/x and f(z)/y.
- With both the substitution, the first expression will be identical to the second expression and the substitution set will be: [a/x, f(z)/y].
- Given: knows(Ram,x). The question is- **Whom does Ram knows?**
- The UNIFY algorithm will search the knowledge base, which could unify with knows(Ram,x).
- UNIFY (Knows(Ram, x), Knows(Ram, Shyam))?{Shyam/x}
- UNIFY (Knows{Ram,x}, Knows{y, Aakash})?{Aakash/x, Ram/y}
- UNIFY (Knows{Ram,x}, Knows{x, Raman})?fails.
- The last one failed because we have used the same variable for two persons at the same time.



1. unify((parents X (father X) (mother bill)), (parents bill (father bill) Y))
Unify first elements and apply substitutions to rest
return { }
return {(mother bill)/Y, bill/X}
2. unify(parents, parents)
3. unify((X (father X) (mother bill)), (bill (father bill) Y))
Unify first elements and apply substitutions to rest
return {bill/X}
return {(mother bill)/Y}
4. unify(X,bill)
5. unify(((father bill) (mother bill)), ((father bill) Y))
Unify first elements and apply substitutions to rest
return { }
return {(mother bill)/Y}
6. unify((father bill), (father bill))
11. unify(((mother bill)), (Y))
Unify first elements and apply substitutions to rest
return { }
return { }
Unify first elements and apply substitutions to rest
return {(mother bill)/Y}
return { }
7. unify(father, father)
8. unify((bill), (bill))
12. unify((mother bill), Y)
13. unify((), ())
Unify first elements and apply substitutions to rest
return { }
return { }
9. unify(bill, bill)
10. unify((), ())

Knowledge Representation (KR) and Reasoning (KRR)

- **Definition:** Knowledge representation is the process of encoding information about the world in a form that a computer can use to solve problems. Reasoning is the process of using the knowledge to draw new conclusions or solve problems.
- It is also a way which describes how we can represent knowledge in artificial intelligence. Knowledge representation is not just storing data into some database, but it also enables an intelligent machine to learn from that knowledge and experiences so that it can behave intelligently like a human.

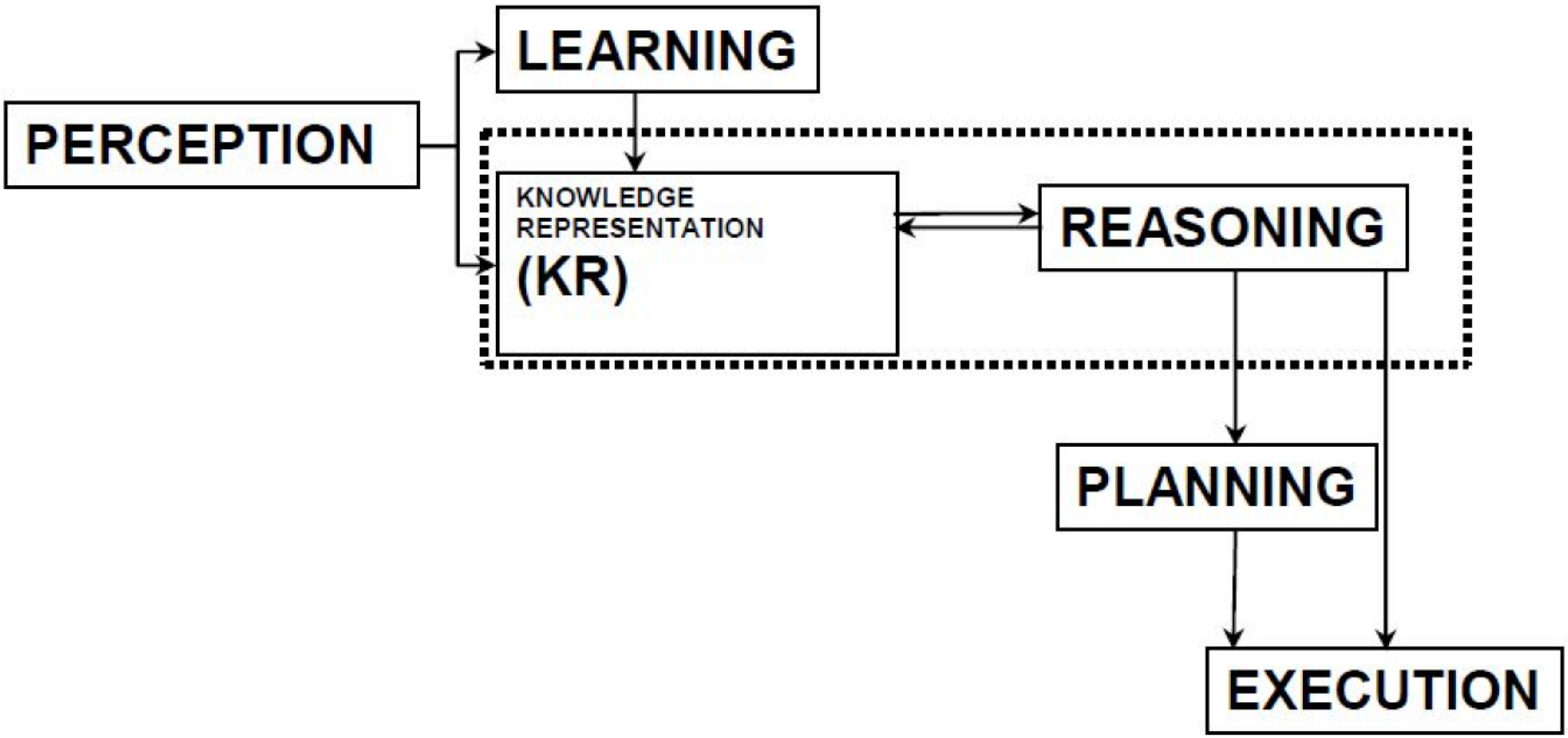
What to Represent?

- **Object:** All the information about the objects in our real world domain. Example: car has wheels, train is a locomotive, guitar contains strings etc.,
- **Events:** refers to something happens in the real world at a particular point of time. Example: war, natural disaster, famine etc.,
- **Performance:** It deals with how humans and other beings and things perform certain actions in different situations. Thus, it helps in understanding the behavior side of the knowledge.
- **Facts:** As the name suggests, this is the knowledge of the factual description of the world. Example: you need oxygen to survive.

Types of Knowledge

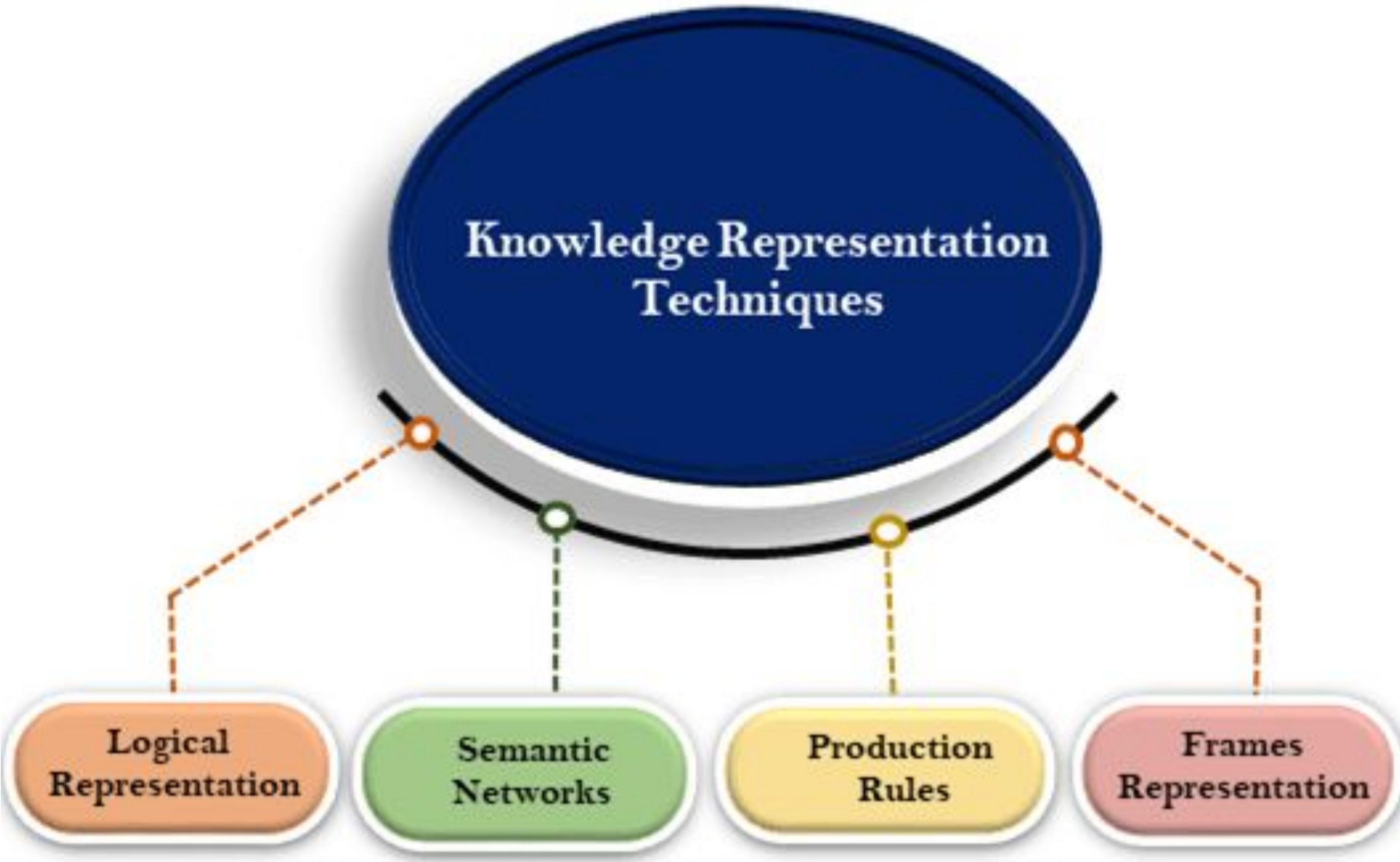
- **Declarative Knowledge:** It is the knowledge that represents the facts, objects, concepts that help us describe the world around us. Thus it deals with the description of something. **Example:** radius of the earth is about 6500 km
- **Procedural Knowledge:** This type of knowledge is more complex than declarative knowledge as it refers to a more complex idea, i.e., how things behave and work. Thus this knowledge is used to accomplish any task using certain procedures, rules, and strategies, making the system using this knowledge work efficiently. **Example:** algorithm for performing long division.
- **Heuristic Knowledge:** The knowledge provided by experts of certain domains, subjects, disciplines, and fields is known as the Heuristic knowledge, which they have been obtained after years of experience.

AI Knowledge Cycle



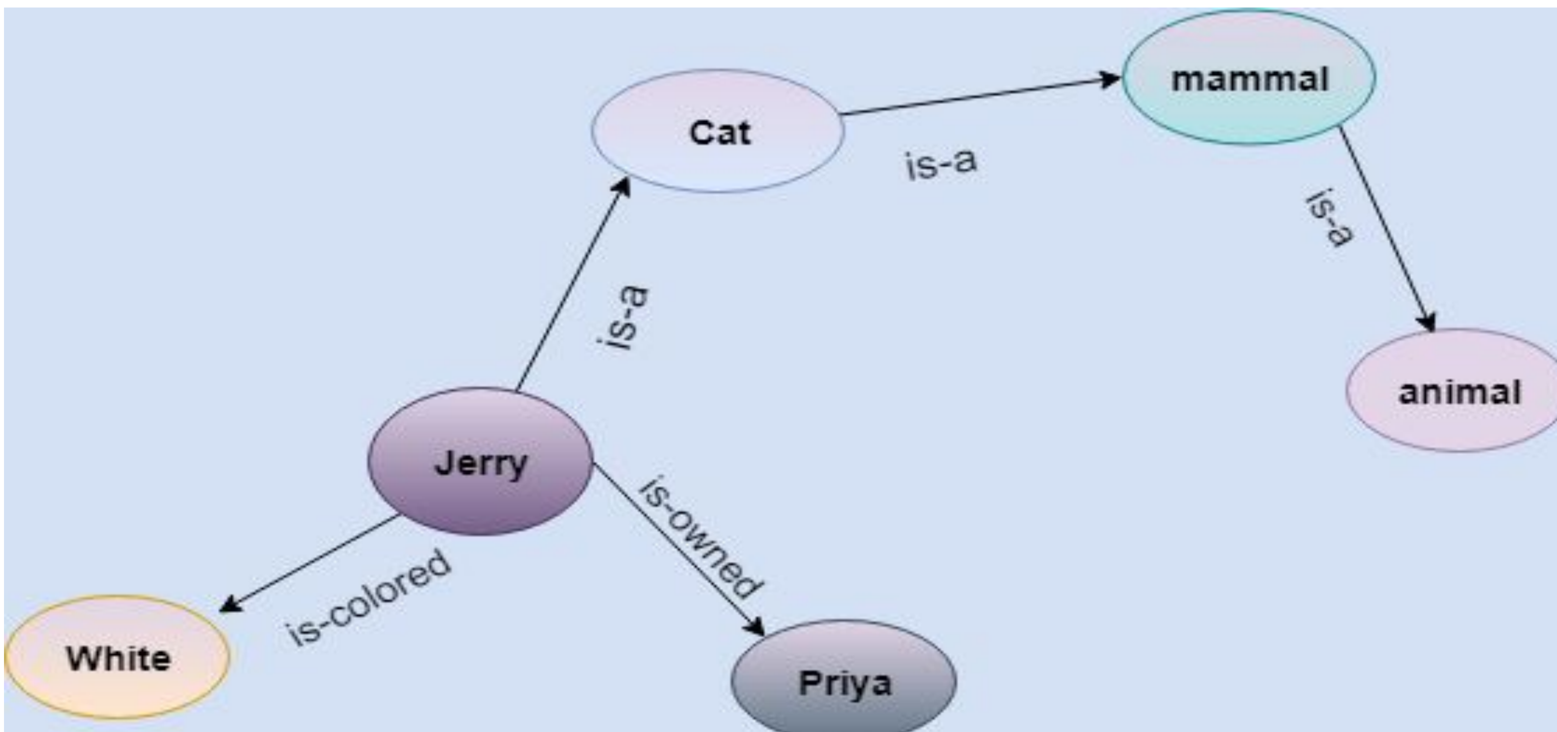
- **Perception Block:** This will help the AI system gain information regarding its surroundings through various sensors, thus making the AI system familiar with its environment and helping it interact with it. These senses can be in the form video, audio, text, time, temperature, or any other sensor-based input.
- **Learning Block:** The knowledge gained will help the AI system to run the deep learning algorithms. These algorithms are written in the learning block, making the AI system transfer the necessary information from the perception block to the learning block for learning (training).
- **Knowledge and Reasoning Block:** these two blocks combined together are responsible for acting like humans. It will go through all the knowledge data and find the relevant whenever it is required.
- **Planning and Execution Block:** These two blocks though independent, can work in tandem. These blocks take the information from the knowledge block and the reasoning block and, based on it, execute certain actions.

Techniques to Knowledge Representation



- **Logical representation:** It is the most basic form of representing knowledge to machines where a well-defined syntax with proper rules is used. This syntax needs to have no ambiguity in its meaning and must deal with prepositions. Thus, this logical form of presentation acts as communication rules and is why it can be best used when representing facts to a machine. Logical Representation can be of two types-
  – **Propositional logic**
  – **Predicate logic**
- **Semantic Network Representation:** Semantic networks are alternative of predicate logic for knowledge representation. In Semantic networks, we can represent our knowledge in the form of graphical networks. This network consists of nodes representing objects and arcs which describe the relationship between those objects.

- **Example for Semantic network representation:**
  - Statements:
    - Jerry is a cat.
    - Jerry is a mammal
    - Jerry is owned by Priya.
    - Jerry is white colored.
    - All Mammals are animal.



- **Frame Representation:** A frame is a record like structure which consists of a collection of attributes and its values to describe an entity in the world. It consists of a collection of slots and slot values. These slots may be of any type and sizes. Slots have names and values which are called facets.
- **Example1:** frame for a book Example2: frame for a person

Slots	**Value**
Title	Artificial Intelligence
Genre	Computer Science
Author	Peter Norvig'
Edition	Third Edition
Year	1996
Page	1152

Slots	**Value**
Name	Peter
Profession	Doctor
Age	25
Marital status	Single
Weight	78

- **Production Rules:** Production rules system consist of (**condition, action**) pairs which mean, "If condition then action". It has mainly three parts:
- The set of production rules
- Working Memory
- The recognize-act-cycle

Types of Reasoning

- **Deductive reasoning:** Deductive reasoning is deducing new information from logically related known information. In deductive reasoning, the truth of the premises guarantees the truth of the conclusion.
  - **Example:**
    - **Premise-1: All the human eats veggies**
    - **Premise-2: Suresh is human.**
    - **Conclusion: Suresh eats veggies.**
- **Inductive Reasoning:** Inductive reasoning is a form of reasoning to arrive at a conclusion using limited sets of facts by the process of generalization. It starts with the series of specific facts or data and reaches to a general statement or conclusion.
  - Example:
    - **Premise: All of the pigeons we have seen in the zoo are white.**
    - **Conclusion: Therefore, we can expect all the pigeons to be white.**

- **Common Sense Reasoning:** Common sense reasoning is an informal form of reasoning, which can be gained through experiences. It relies on good judgment rather than exact logic and operates on **heuristic knowledge** and **heuristic rules.**
  - **Example: If I put my hand in a fire, then it will burn.**
- **Monotonic Reasoning:** In monotonic reasoning, once the conclusion is taken, then it will remain the same even if we add some other information to existing information in our knowledge base. Monotonic reasoning is not useful for the real-time systems, as in real time, facts get changed, so we cannot use monotonic reasoning. It can be used in conventional reasoning systems, and a logic-based system is monotonic.
  - **Example: Earth revolves around the Sun.**
- **Non-monotonic Reasoning:** some conclusions may be invalidated if we add some more information to our knowledge base.
  - **Example: Birds can fly, Penguins cannot fly, Pitty is a bird**
  - So from the above sentences, we can conclude that **Pitty can fly**. If we add “**Pitty is a penguin**", which concludes "**Pitty cannot fly**“.

Reasoning with Uncertain Knowledge

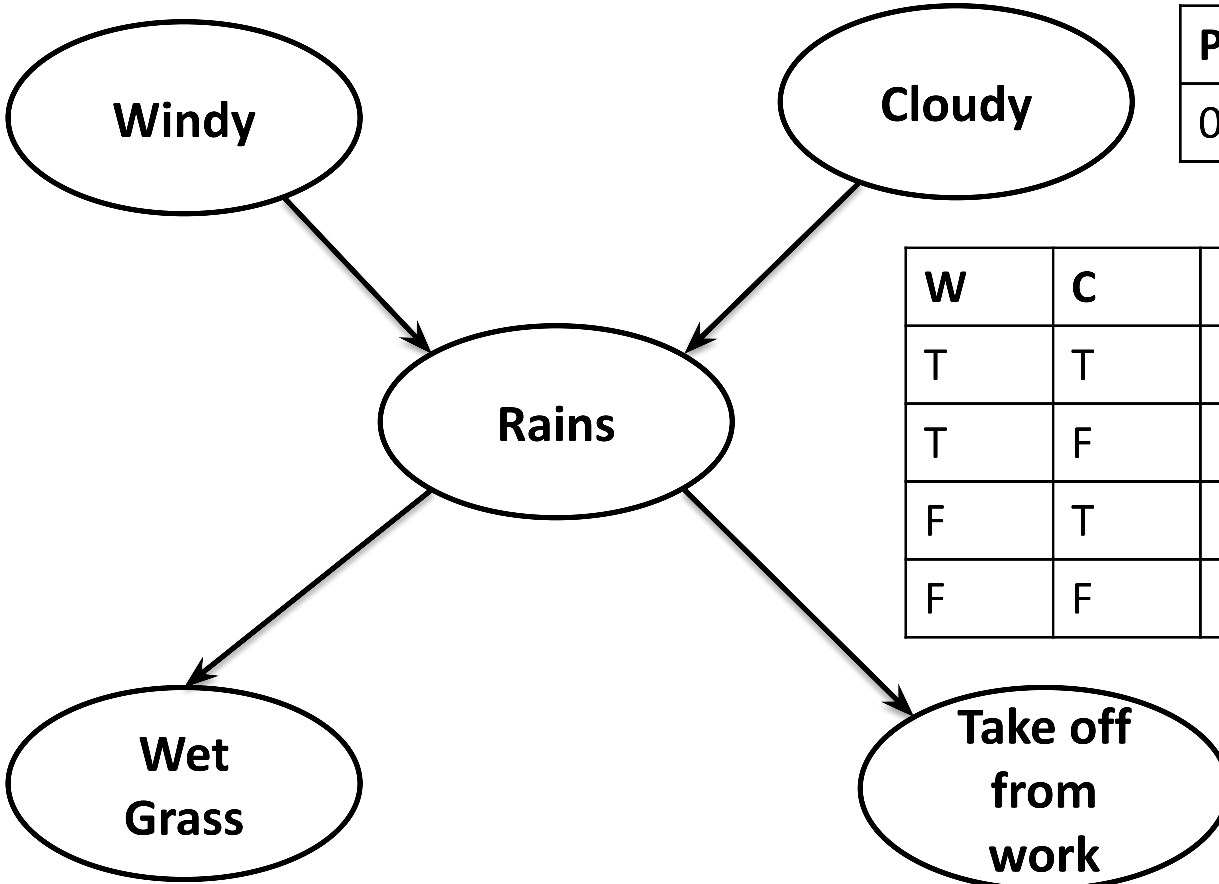
- Reasoning plays a great role in the process of artificial intelligence. Thus reasoning can be defined as the logical process of drawing conclusion, making predictions or constructing approaches towards particular thought with the help of existing knowledge.
- In real life, it is not always possible to determine the state of the environment as it might not be clear. Due to partially observable or non-deterministic environments, agents may need to handle uncertainty and deal with: Uncertain data, Uncertain knowledge, Inference.
  - **Example**: In case of Medical diagnosis consider the rule Toothache = Cavity. This is not complete as not all patients having toothache have cavities. So we can write a more generalized rule Toothache = Cavity V Gum problems V Abscess… To make this rule complete, we will have to list all the possible causes of toothache. But this is not feasible due to laziness, Theoretical ignorance and Practical ignorance.

Ways to handle uncertainty

- **Probabilistic Reasoning:** it is a way of knowledge representation where we apply the concept of probability to indicate the uncertainty in knowledge. In probabilistic reasoning, we combine probability theory with logic to handle the uncertainty.
- Bayesian belief network is key computer technology for dealing with probabilistic events and to solve a problem which has uncertainty.
- A Bayesian network is a probabilistic graphical model which represents a set of variables and their conditional dependencies using a directed acyclic graph.
- Bayesian networks are probabilistic, because these networks are built from a **probability distribution**, and also use probability theory for prediction

P(W)
0.001

P(C)
0.002



W	C	P(R)
T	T	0.95
T	F	0.95
F	T	0.29
F	F	0.01

R	P(WG)
T	0.95
F	0.05

R	P(O)
T	0.7
F	0.01

Find the probability of wet grass:

$$P(WG) = P(WG|R) * P(R) + P(WG|\neg R) * P(\neg R) \text{-----------} (1)$$

$$= 0.95 * P(R) + 0.05 * P(\neg R)$$

$$P(R) = P(R|W,C)*P(W \wedge C)+P(R|\neg W,C)*P(\neg W \wedge C)+ P(R|W,\neg C)*P(W \wedge \neg C)+ P(R|\neg W,\neg C) *P(\neg W \wedge \neg C) \text{-----}(2)$$

$$=0.95*0.001*0.002+0.29*0.999*0.002+0.95*0.001*0.998+0.01* 0.999*0.998$$

P(R) = 0.00252.

Similarly

P(¬R) = 0.99744

P(WG) = 0.0521