Artificial Intelligence

Pushkar Harshe

What is AI

- “The Science and Engineering of making intelligent Machine Especially intelligent computer Program” ..John Makarthy
- AI Accomplish
- Human Learning
- Human Decision
- Problem solving

Contributing Areas For Building AI

- 1) Computer Science
- 2)Philosophy
- 3)Neuron Science
- 4)Biology
- 5)Maths
- 6) Sociology
- 7) Psychology

Use of AI

- Gamming
- NLP
- Expert System
- Vision System
- Speech Recognition
- Handwriting Recognition
- Intelligent Robot

Advancement in AI

- Machine Learning
- Case Based Learning
- Scheduling
- Data mining web crawler
- NL Understanding and Translation
- Virtual Reality
- Simulator

Types of Intelligence

Intelligence	Description	Example
Linguistic Intelligence	Ability to Speak and Phonology syntax (Grammar) and meaning	Narrators/ Orators
Musical Intelligence	Sound Pitch Rhythm	Musician/Singer/Composers
Logical/Mathematical Intelligence	Understand complex and abstract idea	Scientist
Visual / Spatial Intelligence	Visual Change 2d/3d image its rotation	Map Readers Physicist Astronauts
Body / Kinesthetic Intelligence	Motor skill to use some or entire body to solve Problem	Players / Dancers
Intra Personal Intelligence	Ones feeling , intension motivation	Buddha
Inter personal Intelligence	Distinguish among people beliefs feelings etc	Mass Communicator

Intelligence is composed of

- Reasoning
- Learning
- Problem Solving
- Perception
- Linguistic Intelligence

- **Types Of Reasoning**

Inductive Reasoning	Deductive Reasoning
Specific Observations to Make Broad	Start with general statement examine to reach specific logical conclusion
Statements Even if all premises are true it allow conclusion to be false	If something is true in a class statement then all members are true
Neema is Teacher all teachers are studious hence Neema is studious	All Men above 60 yrs are grand father . Pushkar is 65 Hence he is Grandfather

Learning types

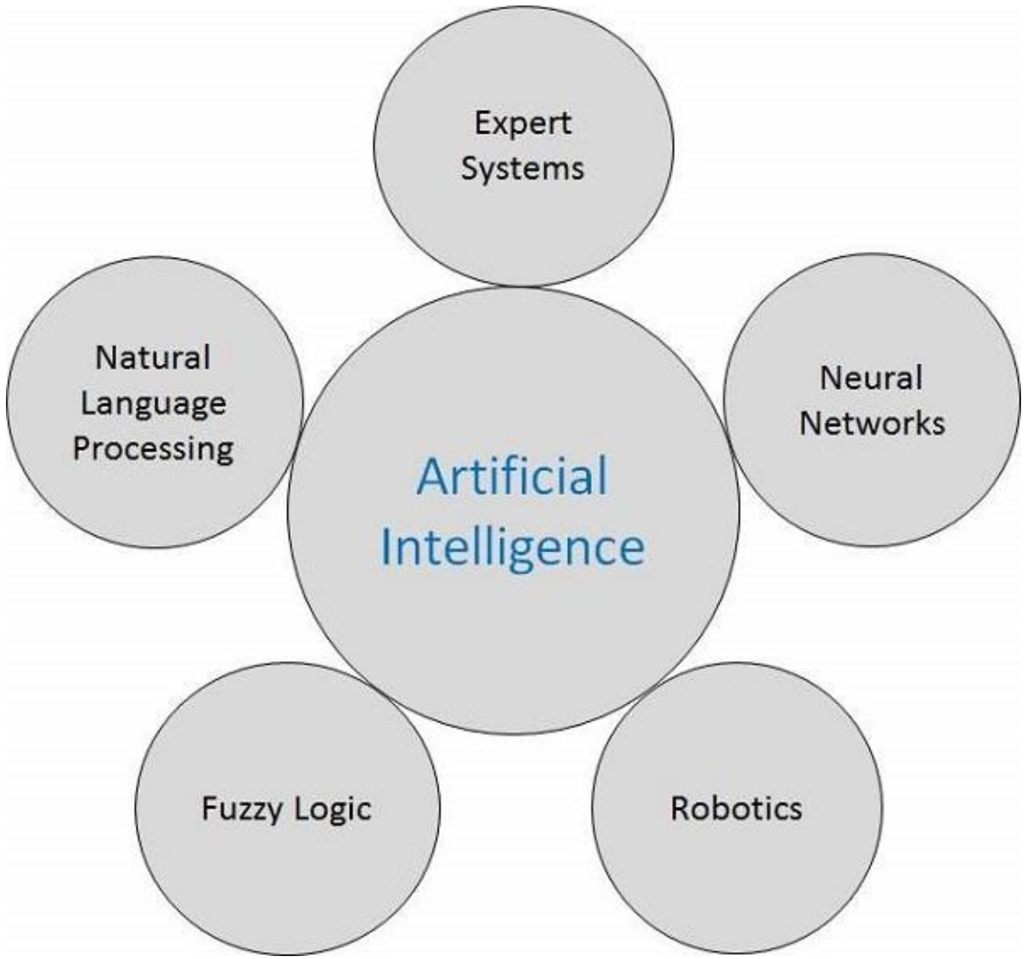
- **Auditory Learning** − It is learning by listening and hearing. For example, students listening to recorded audio lectures.
- o **Episodic Learning** − To learn by remembering sequences of events that one has witnessed or experienced. This is linear and orderly.
- o **Motor Learning** − It is learning by precise movement of muscles. For example, picking objects, Writing, etc.
- o **Observational Learning** − To learn by watching and imitating others. For example, child tries to learn by mimicking her parent.
- o **Perceptual Learning** − It is learning to recognize stimuli that one has seen before. For example, identifying and classifying objects and situations.
- o **Relational Learning** − It involves learning to differentiate among various stimuli on the basis of relational properties, rather than absolute properties. For Example, Adding 'little less' salt at the time of cooking potatoes that came up salty last time, when cooked with adding say a tablespoon of salt.
- o **Spatial Learning** − It is learning through visual stimuli such as images, colors, maps, etc. For Example, A person can create roadmap in mind before actually following the road.
- o **Stimulus-Response Learning** − It is learning to perform a particular behavior when a certain stimulus is present. For example, a dog raises its ear on hearing doorbell.

- **Problem Solving** − It is the process in which one perceives and tries to arrive at a desired solution from a present situation by taking some path, which is blocked by known or unknown hurdles.
- Problem solving also includes **decision making**, which is the process of selecting the best suitable alternative out of multiple alternatives to reach the desired goal are available.
- · **Perception** − It is the process of acquiring, interpreting, selecting, and organizing sensory information.
- Perception presumes **sensing**. In humans, perception is aided by sensory organs. In the domain of AI, perception mechanism puts the data acquired by the sensors together in a meaningful manner.
- · **Linguistic Intelligence** − It is one's ability to use, comprehend, speak, and write the verbal and written language. It is important in interpersonal communication.

Difference between Human and Machine Intelligence

- Humans perceive by patterns whereas the machines perceive by set of rules and data.
- · Humans store and recall information by patterns, machines do it by searching algorithms. For example, the number 40404040 is easy to remember, store, and recall as its pattern is simple.
- · Humans can figure out the complete object even if some part of it is missing or distorted; whereas the machines cannot do it correctly.
-

AI - Research Areas



AI - Research Areas

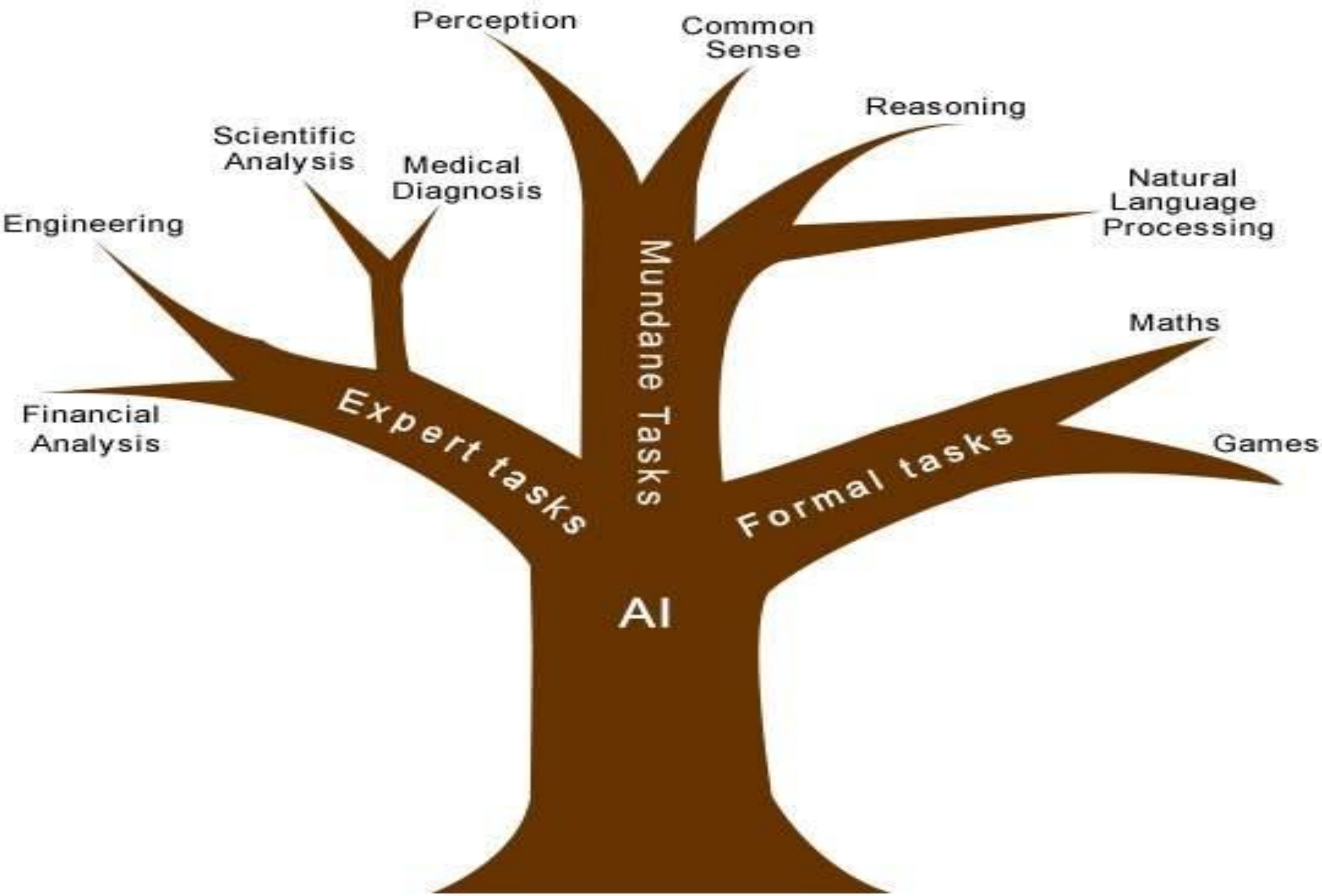
- Speech and Voice Recognition

Speech Recognition	Voice Recognition
The speech recognition aims at understanding and comprehending **WHAT** was spoken.	The objective of voice recognition is to recognize **WHO** is speaking.
It is used in hand-free computing, map, or menu navigation.	It is used to identify a person by analysing its tone, voice pitch, and accent, etc.
Machine does not need training for Speech Recognition as it is not speaker dependent.	This recognition system needs training as it is person oriented.
Speaker independent Speech Recognition systems are difficult to develop.	Speaker dependent Speech Recognition systems are comparatively easy to develop.

Real Life Applications of AI Research Areas

Sr.No.	Research Areas	Example
1	**Expert Systems** Examples – Flight-tracking systems, Clinical systems.	
2	**Natural Language Processing** Examples: Google Now feature, speech recognition, Automatic voice output.	
3	**Neural Networks** Examples – Pattern recognition systems such as face recognition, character recognition, handwriting recognition.	
4	**Robotics** Examples – Industrial robots for moving, spraying, painting, precision checking, drilling, cleaning, coating, carving, etc.	
5	**Fuzzy Logic Systems**	

Task Classification of AI

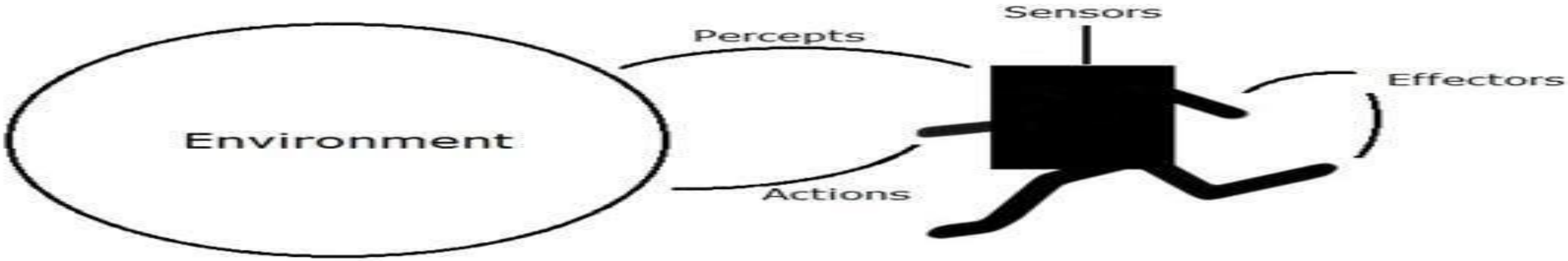


Task Domains of Artificial Intelligence

Mundane (Ordinary) Tasks	**Formal Tasks**	**Expert Tasks**
Perception •Computer Vision •Speech, Voice	•Mathematics •Geometry •Logic •Integration and Differentiation	•Engineering •Fault Finding •Manufacturing •Monitoring
Natural Language Processing •Understanding •Language Generation •Language Translation	Games •Go •Chess (Deep Blue) •Checkers	Scientific Analysis
Common Sense	Verification	Financial Analysis
Reasoning	Theorem Proving	Medical Diagnosis
Planing		Creativity
Robotics •Locomotive		

AI - Agents & Environments

- What are Agent and Environment?
- An **agent** is anything that can perceive its environment through **sensors** and acts upon that environment through **effectors.**
- · A **human agent** has sensory organs such as eyes, ears, nose, tongue and skin parallel to the sensors, and other organs such as hands, legs, mouth, for effectors.
- · A **robotic agent** replaces cameras and infrared range finders for the sensors, and various motors and actuators for effectors.
- · A **software agent** has encoded bit strings as its programs and actions.



- Agent Terminology
- · **Performance Measure of Agent** – It is the criteria, which determines how successful an agent is.
- · **Behavior of Agent** – It is the action that agent performs after any given sequence of percepts.
- · **Percept** – It is agent’s perceptual inputs at a given instance.
- · **Percept Sequence** – It is the history of all that an agent has perceived till date.
- · **Agent Function** – It is a map from the precept sequence to an action.

Rationality

Rationality is nothing but status of being reasonable, sensible, and having good sense of judgment.

Rationality is concerned with expected actions and results depending upon what the agent has perceived. Performing actions with the aim of obtaining useful information is an important part of rationality.

What is Ideal Rational Agent?

An ideal rational agent is the one, which is capable of doing expected actions to maximize its performance measure, on the basis of −

Its percept sequence

Its built-in knowledge base

Rationality of an agent depends on the following four factors −

- The **performance measures**, which determine the degree of success.
- Agent’s **Percept Sequence** till now.
- The agent’s **prior knowledge about the environment**.
- The **actions** that the agent can carry out.

A rational agent always performs right action, where the right action means the action that causes the agent to be most successful in the given percept sequence. The problem the agent solves is characterized by Performance Measure, Environment, Actuators, and Sensors (PEAS).

The Structure of Intelligent Agents

Agent’s structure can be viewed as –

Agent = Architecture + Agent Program

Architecture = the machinery that an agent executes on.

Agent Program = an implementation of an agent function.

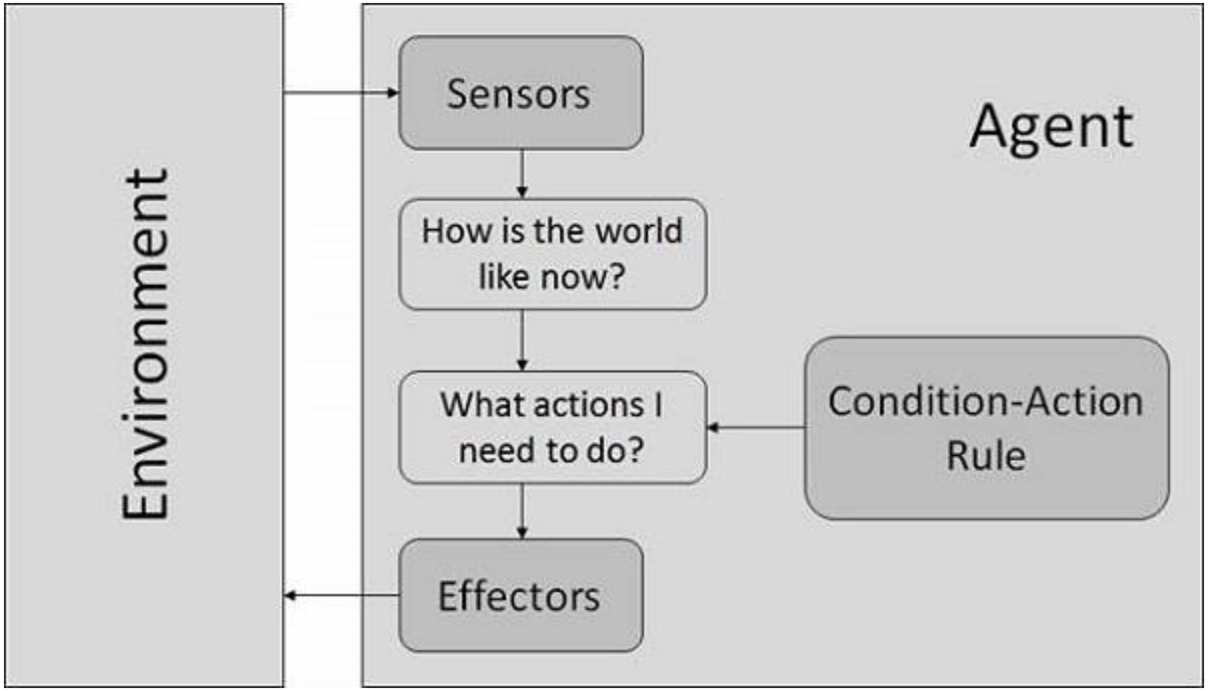
Simple Reflex Agents

They choose actions only based on the current percept.

They are rational only if a correct decision is made only on the basis of current precept.

Their environment is completely observable.

Condition-Action Rule − It is a rule that maps a state (condition) to an action.



Model Based Reflex Agents

They use a model of the world to choose their actions. They maintain an internal state.

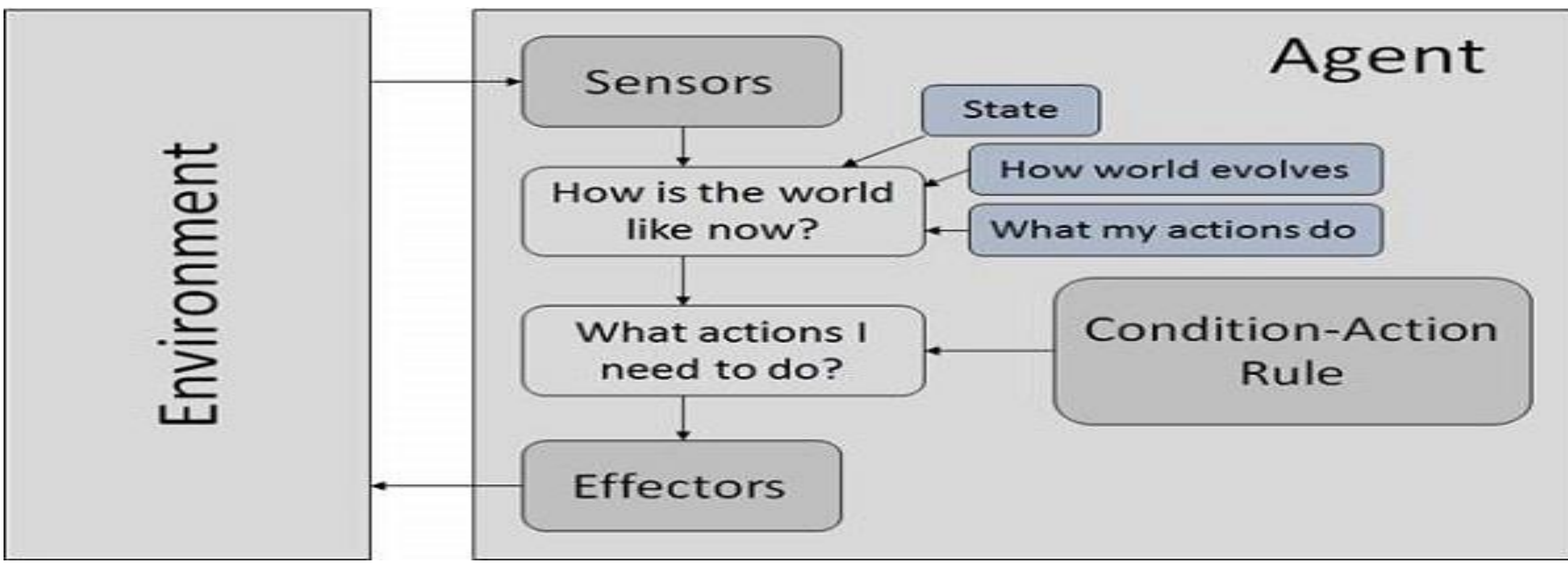
Model − The knowledge about “how the things happen in the world”.

Internal State − It is a representation of unobserved aspects of current state depending on percept history.

Updating the state requires the information about −

How the world evolves.

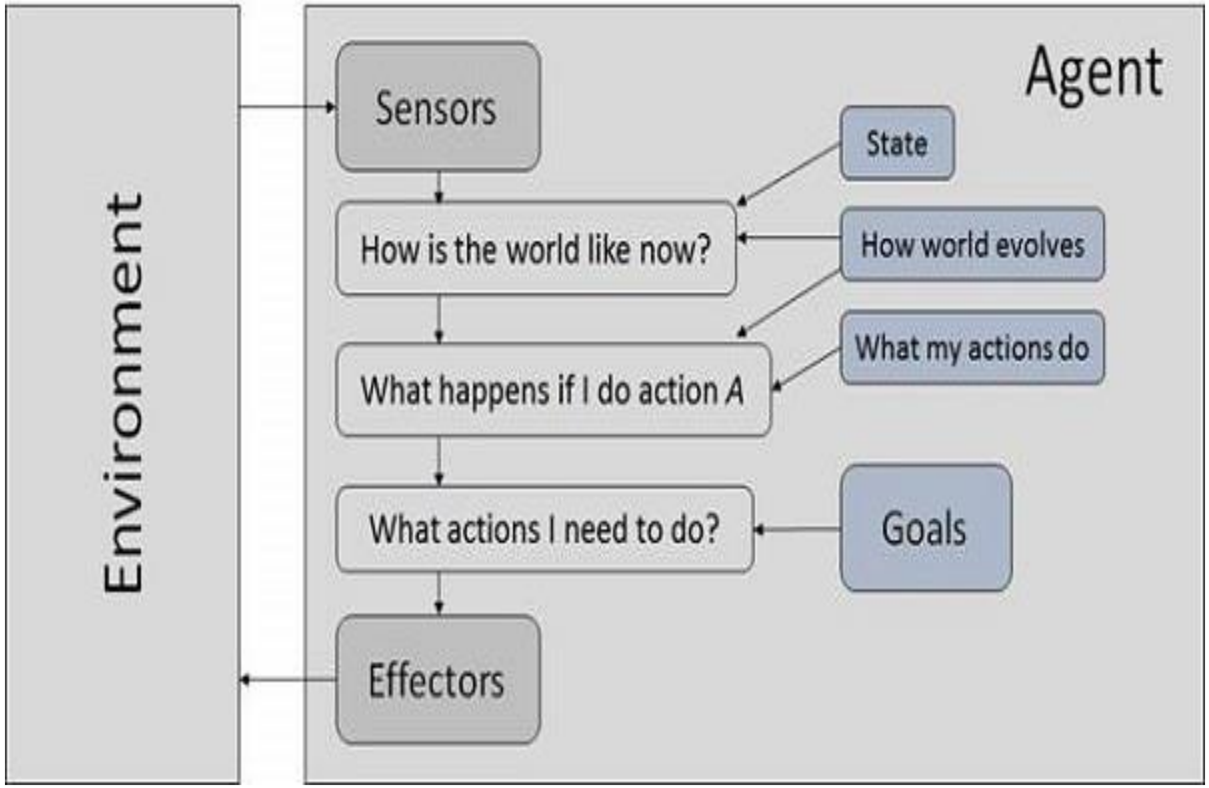
How the agent’s actions affect the world.



Goal Based Agents

They choose their actions in order to achieve goals. Goal-based approach is more flexible than reflex agent since the knowledge supporting a decision is explicitly modeled, thereby allowing for modifications.

Goal − It is the description of desirable situations.

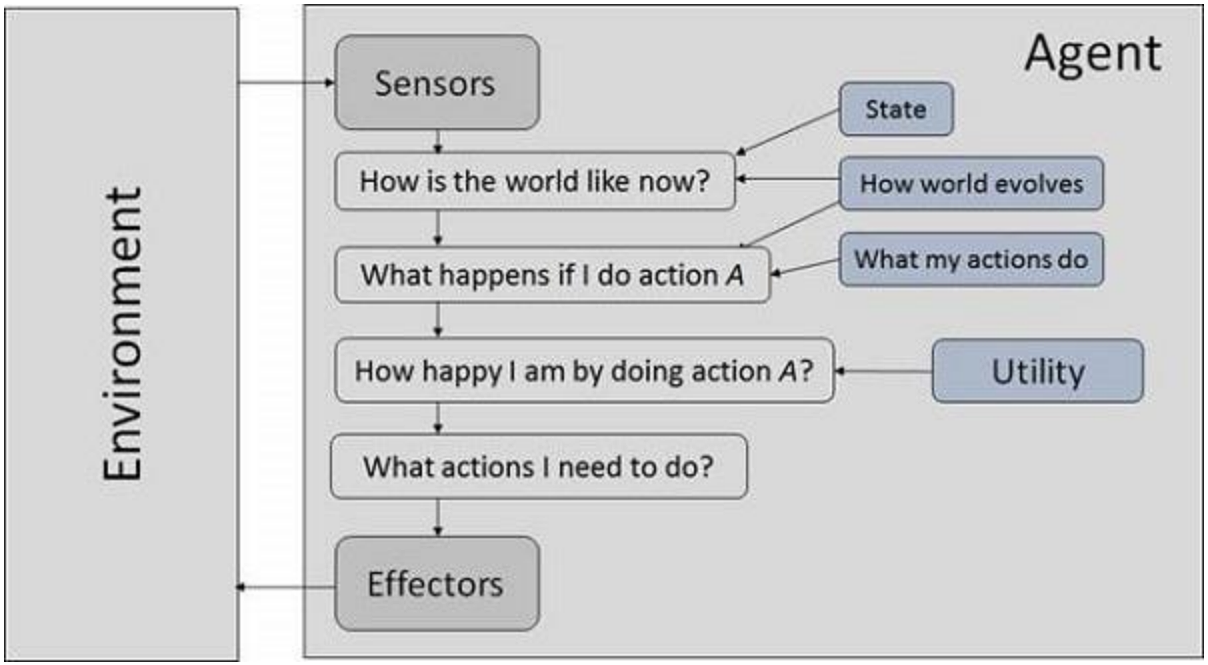


Utility Based Agents

They choose actions based on a preference (utility) for each state. Goals are inadequate when −

· There are conflicting goals, out of which only few can be achieved.

· Goals have some uncertainty of being achieved and you need to weigh likelihood of success against the importance of a goal.



Nature of Environments

Some programs operate in the entirely artificial environment confined to keyboard input, database, computer file systems and character output on a screen.

In contrast, some software agents (software robots or soft bots) exist in rich, unlimited soft bots domains. The simulator has a very detailed, complex environment. The software agent needs to choose from a long array of actions in real time. A soft bot designed to scan the online preferences of the customer and show interesting items to the customer works in the real as well as an artificial environment.

The most famous artificial environment is the Turing Test environment, in which one real and other artificial agents are tested on equal ground. This is a very challenging environment as it is highly difficult for a software agent to perform as well as a human.

Turing Test

The success of an intelligent behaviour of a system can be measured with Turing Test.

Two persons and a machine to be evaluated participate in the test. Out of the two persons, one plays the role of the tester. Each of them sits in different rooms. The tester is unaware of who is machine and who is a human. He interrogates the questions by typing and sending them to both intelligences, to which he receives typed responses.

This test aims at fooling the tester. If the tester fails to determine machine's response from the human response, then the machine is said to be intelligent.

Properties of Environment

- **Discrete / Continuous** − If there are a limited number of distinct, clearly defined, states of the environment, the environment is discrete (For example, chess); otherwise it is continuous (For example, driving).
- · **Observable / Partially Observable** − If it is possible to determine the complete state of the environment at each time point from the percepts it is observable; otherwise it is only partially observable.
- · **Static / Dynamic** − If the environment does not change while an agent is acting, then it is static; otherwise it is dynamic.
- · **Single agent / Multiple agents** − The environment may contain other agents which may be of the same or different kind as that of the agent.
- · **Accessible / Inaccessible** − If the agent's sensory apparatus can have access to the complete state of the environment, then the environment is accessible to that agent.
- · **Deterministic / Non-deterministic** − If the next state of the environment is completely determined by the current state and the actions of the agent, then the environment is deterministic; otherwise it is non-deterministic.
- · **Episodic / Non-episodic** − In an episodic environment, each episode consists of the agent perceiving and then acting. The quality of its action depends just on the episode itself. Subsequent episodes do not depend on the actions in the previous episodes. Episodic environments are much simpler because the agent does not need to think ahead.
-

AI - Popular Search Algorithms

Searching is the universal technique of problem solving in AI. There are some single-player games such as tile games, Sudoku, crossword, etc. The search algorithms help you to search for a particular position in such games.

Single Agent Path finding Problems

The games such as 3X3 eight-tile, 4X4 fifteen-tile, and 5X5 twenty four tile puzzles are single-agent-path-finding challenges. They consist of a matrix of tiles with a blank tile. The player is required to arrange the tiles by sliding a tile either vertically or horizontally into a blank space with the aim of accomplishing some objective.

The other examples of single agent path finding problems are Travelling Salesman Problem, Rubik's Cube, and Theorem Proving.

Search Terminology

- **Problem Space** − It is the environment in which the search takes place. (A set of states and set of operators to change those states)
- **Problem Instance** − It is Initial state + Goal state.
- **Problem Space Graph** − It represents problem state. States are shown by nodes and operators are shown by edges.
- **Depth of a problem** − Length of a shortest path or shortest sequence of operators from Initial State to goal state.
- **Space Complexity** − The maximum number of nodes that are stored in memory.
- **Time Complexity** − The maximum number of nodes that are created.
- **Admissibility** − A property of an algorithm to always find an optimal solution.
- **Branching Factor** − The average number of child nodes in the problem space graph.
- **Depth** − Length of the shortest path from initial state to goal state.

Brute-Force Search Strategies

Brute-Force Search Strategies

They are most simple, as they do not need any domain-specific knowledge. They work fine with small number of possible states.

Requirements −

State description

A set of valid operators

Initial state

Goal state description

Breadth-First Search

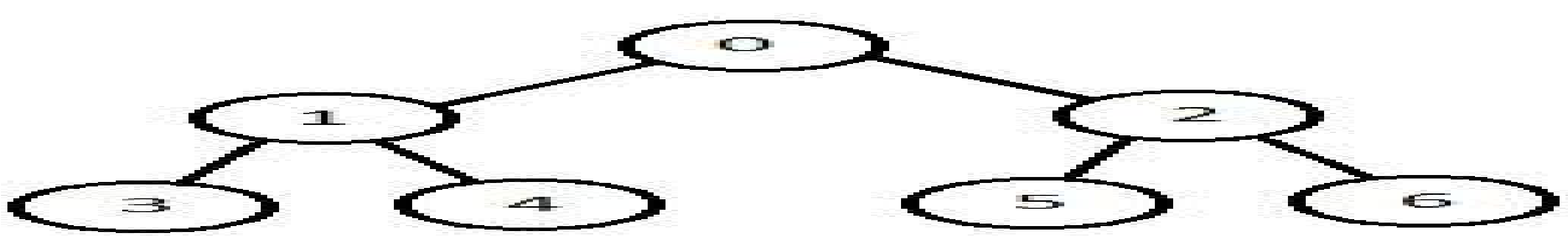
It starts from the root node, explores the neighbouring nodes first and moves towards the next level neighbours. It generates one tree at a time until the solution is found. It can be implemented using FIFO queue data structure. This method provides shortest path to the solution.

If **branching factor** (average number of child nodes for a given node) = b and depth = d, then number of nodes at level d = b^d.

The total no of nodes created in worst case is $b + b^2 + b^3 + … + b^d$.

Disadvantage − Since each level of nodes is saved for creating next one, it consumes a lot of memory space. Space requirement to store nodes is exponential.

Its complexity depends on the number of nodes. It can check duplicate nodes.



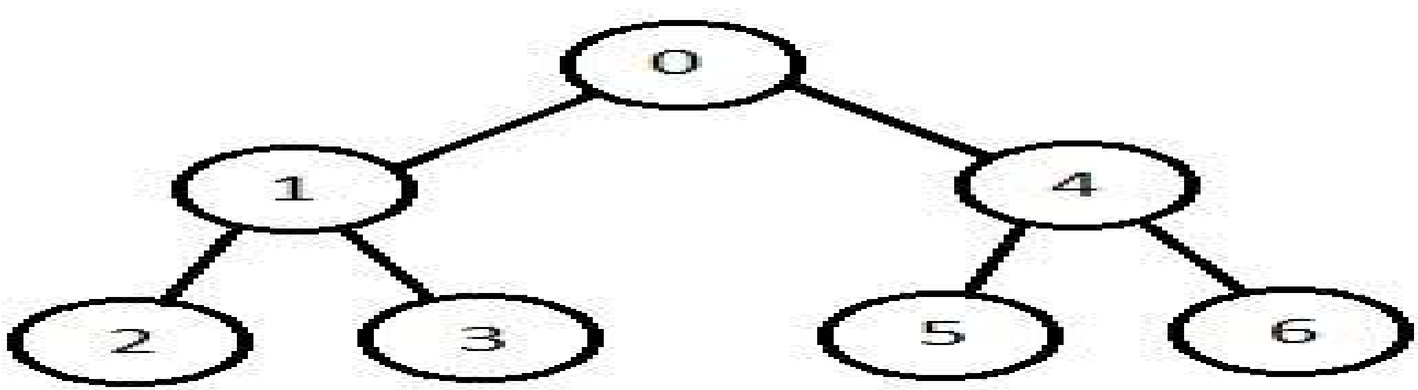
Depth-First Search

It is implemented in recursion with LIFO stack data structure. It creates the same set of nodes as Breadth-First method, only in the different order.

As the nodes on the single path are stored in each iteration from root to leaf node, the space requirement to store nodes is linear. With branching factor *b*and depth as *m*, the storage space is *bm.*

Disadvantage − This algorithm may not terminate and go on infinitely on one path. The solution to this issue is to choose a cut-off depth. If the ideal cut-off is *d*, and if chosen cut-off is lesser than *d*, then this algorithm may fail. If chosen cut-off is more than *d*, then execution time increases.

Its complexity depends on the number of paths. It cannot check duplicate nodes.



Bidirectional Search

It searches forward from initial state and backward from goal state till both meet to identify a common state.

The path from initial state is concatenated with the inverse path from the goal state. Each search is done only up to half of the total path.

Uniform Cost Search

Sorting is done in increasing cost of the path to a node. It always expands the least cost node.

It is identical to Breadth First search if each transition has the same cost.

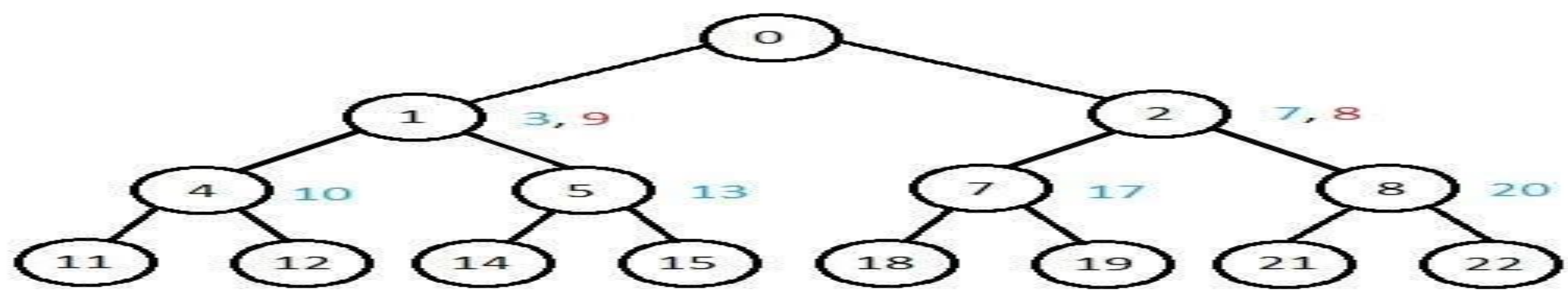
It explores paths in the increasing order of cost.

Disadvantage − There can be multiple long paths with the cost ≤ C*. Uniform Cost search must explore them all.

Iterative Deepening Depth-First Search

It performs depth-first search to level 1, starts over, executes a complete depth-first search to level 2, and continues in such way till the solution is found.

It never creates a node until all lower nodes are generated. It only saves a stack of nodes. The algorithm ends when it finds a solution at depth *d*. The number of nodes created at depth *d* is b^d and at depth *d-1* is b^{d-1}.



Comparison of Various Algorithms Complexities

Let us see the performance of algorithms based on various criteria −

Criterion	Breadth First	Depth First	Bidirectional	Uniform Cost	Interactive Deepening
Time	b^d	b^m	$b^{d/2}$	b^d	b^d
Space	b^d	b^m	$b^{d/2}$	b^d	b^d
Optimality	Yes	No	Yes	Yes	Yes
Completeness	Yes	No	Yes	Yes	Yes

What is Fuzzy Logic?

Fuzzy Logic (FL) is a method of reasoning that resembles human reasoning. The approach of FL imitates the way of decision making in humans that involves all intermediate possibilities between digital values YES and NO.

The conventional logic block that a computer can understand takes precise input and produces a definite output as TRUE or FALSE, which is equivalent to human's YES or NO.

The inventor of fuzzy logic, Lotfi Zadeh, observed that unlike computers, the human decision making includes a range of possibilities between YES and NO, such as −

CERTAINLY YES

POSSIBLY YES

CANNOT SAY

POSSIBLY NO

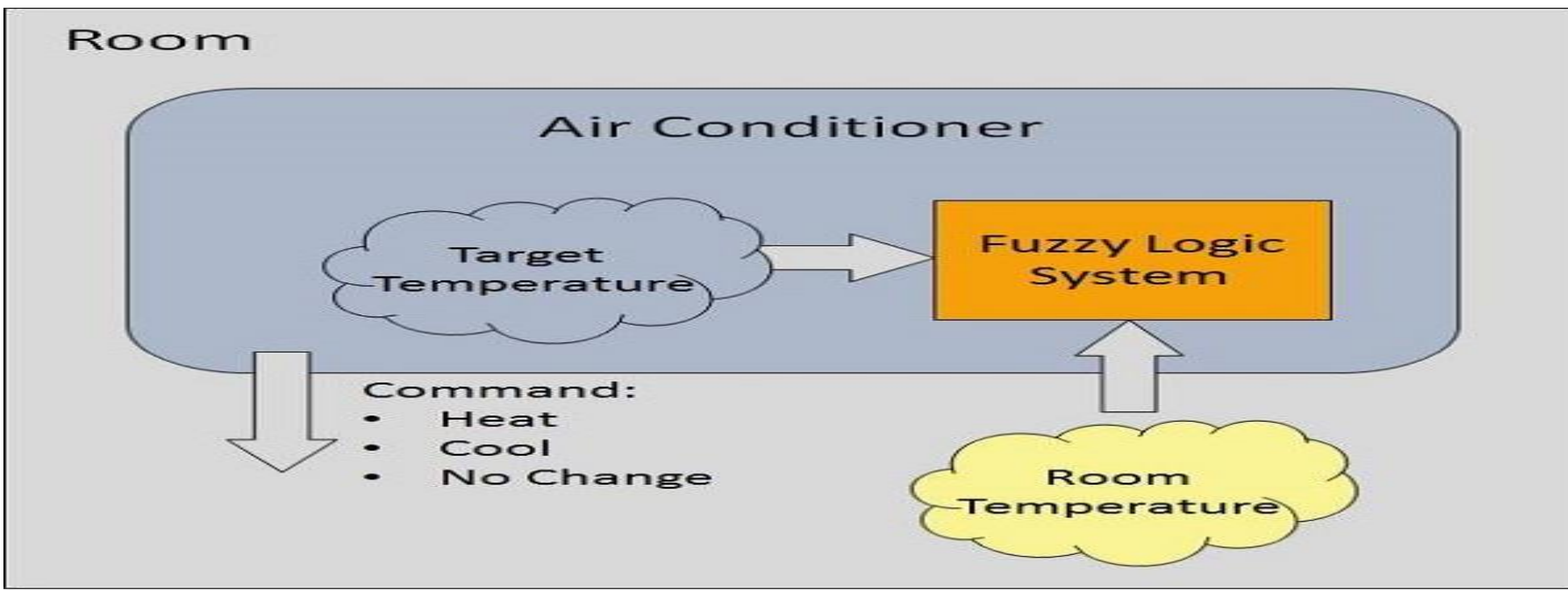
CERTAINLY NO

Example of a Fuzzy Logic System

Let us consider an air conditioning system with 5-level fuzzy logic system. This system adjusts the temperature of air conditioner by comparing the room temperature and the target temperature value.

Algorithm

1 Define linguistic variables and terms.
2 Construct membership functions for them.
3 Construct knowledge base of rules.
4 Convert crisp data into fuzzy data sets using membership functions. (fuzzification)
5 Evaluate rules in the rule base. (Inference Engine)
6 Combine results from each rule. (Inference Engine)
7 Convert output data into non-fuzzy values. (de fuzzification)



Logic Development

Step 1: Define linguistic variables and terms

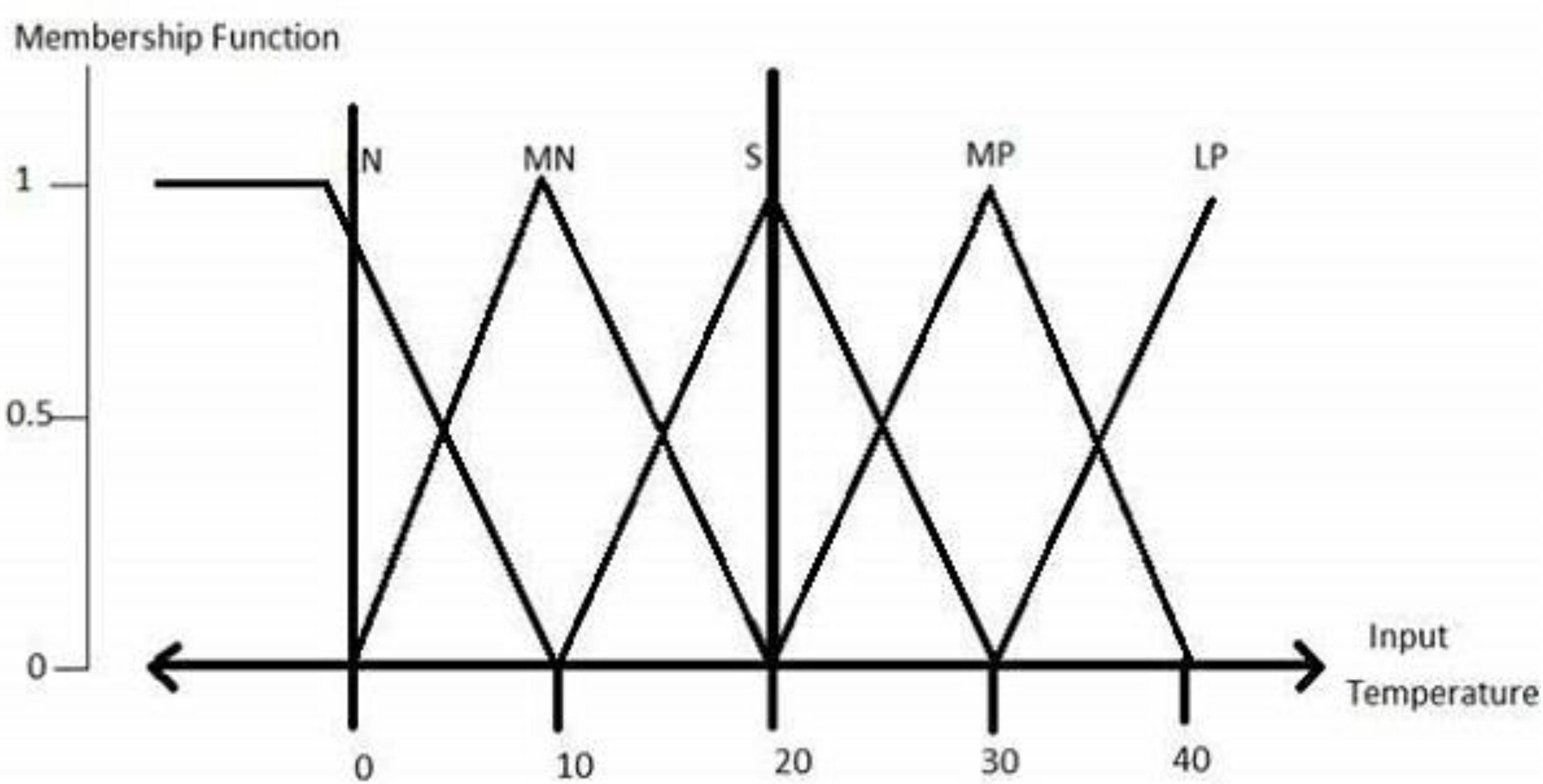
Linguistic variables are input and output variables in the form of simple words or sentences. For room temperature, cold, warm, hot, etc., are linguistic terms.

Temperature (t) = {very-cold, cold, warm, very-warm, hot}

Every member of this set is a linguistic term and it can cover some portion of overall temperature values.

Step 2: Construct membership functions for them

The membership functions of temperature variable are as shown



Step3: Construct knowledge base rules

Create a matrix of room temperature values versus target temperature values that an air conditioning system is expected to provide.

RoomTemp. /Target	Very_Cold	Cold	Warm	Hot	Very_Hot
Very_Cold	No_Change	Heat	Heat	Heat	Heat
Cold	Cool	No_Change	Heat	Heat	Heat
Warm	Cool	Cool	No_Change	Heat	Heat
Hot	Cool	Cool	Cool	No_Change	Heat
Very_Hot	Cool	Cool	Cool	Cool	No_Change

Build a set of rules into the knowledge base in the form of IF-THEN-ELSE structures

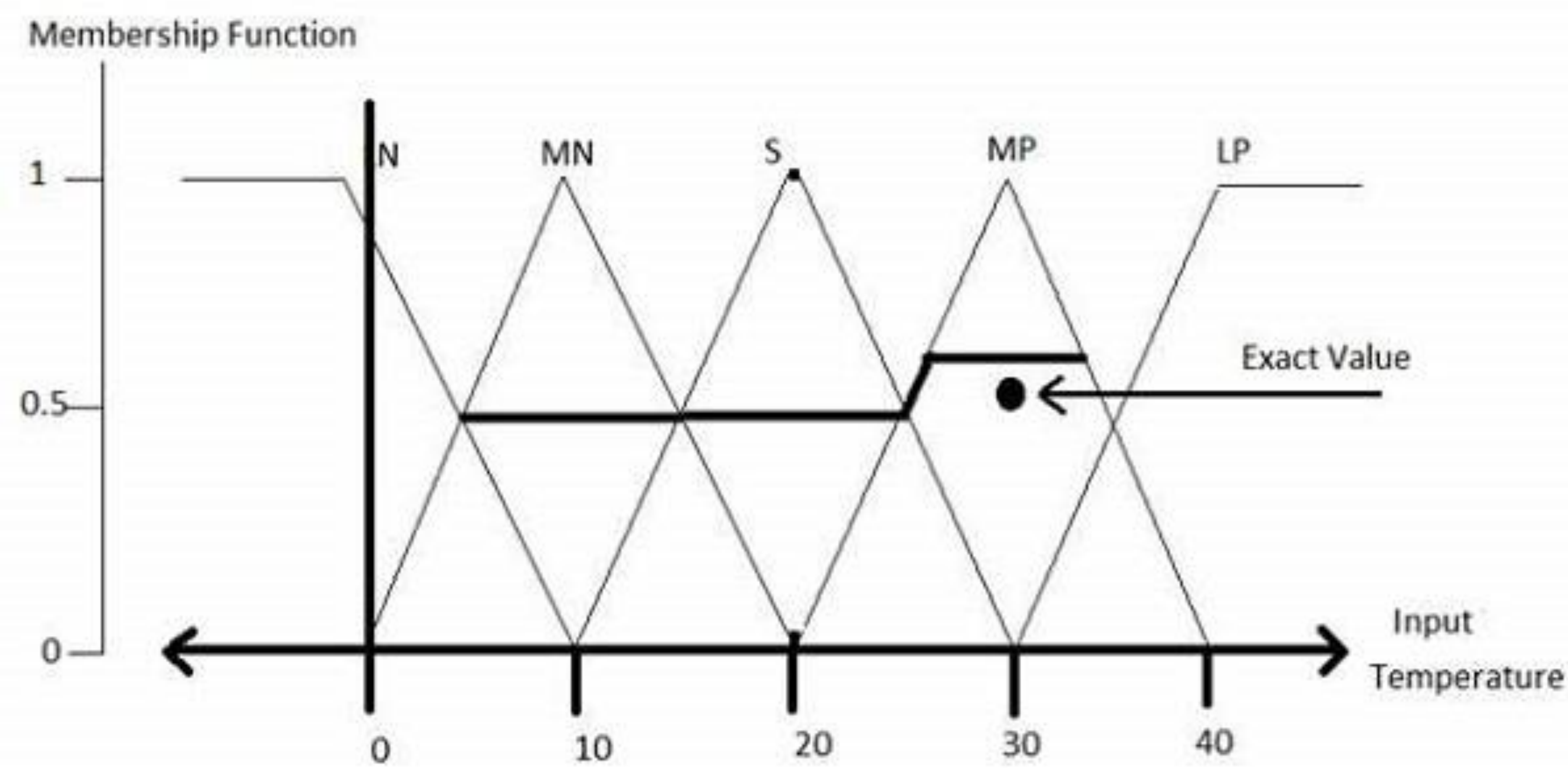
Sr. No.	Condition	Action
1	IF temperature=(Cold OR Very_Cold) AND target=Warm THEN	Heat
2	IF temperature=(Hot OR Very_Hot) AND target=Warm THEN	Cool
3	IF (temperature=Warm) AND (target=Warm) THEN	No_Change

Step 4: Obtain fuzzy value

Fuzzy set operations perform evaluation of rules. The operations used for OR and AND are Max and Min respectively. Combine all results of evaluation to form a final result. This result is a fuzzy value.

Step 4: Obtain fuzzy value

Fuzzy set operations perform evaluation of rules. The operations used for OR and AND are Max and Min respectively. Combine all results of evaluation to form a final result. This result is a fuzzy value.



Application Areas of Fuzzy Logic

The key application areas of fuzzy logic are as given −

Automotive Systems

Automatic Gearboxes

Four-Wheel Steering

Vehicle environment control

Consumer Electronic Goods

Hi-Fi Systems

Photocopiers

Still and Video Cameras

Television

Domestic Goods

Microwave Ovens

Refrigerators

Toasters

Vacuum Cleaners

Washing Machines

Environment Control

Air Conditioners/Dryers/Heaters

Humidifiers

Advantages of FLSs

- Mathematical concepts within fuzzy reasoning are very simple.
- You can modify a FLS by just adding or deleting rules due to flexibility of fuzzy logic.
- Fuzzy logic Systems can take imprecise, distorted, noisy input information.
- FLSs are easy to construct and understand.
- Fuzzy logic is a solution to complex problems in all fields of life, including medicine, as it resembles human reasoning and decision making.

Disadvantages of FLSs

There is no systematic approach to fuzzy system designing.
They are understandable only when simple.
They are suitable for the problems which do not need high accuracy.

AI - Natural Language Processing

Natural Language Processing (NLP) refers to AI method of communicating with an intelligent systems using a natural language such as English.

Processing of Natural Language is required when you want an intelligent system like robot to perform as per your instructions, when you want to hear decision from a dialogue based clinical expert system, etc.

The field of NLP involves making computers to perform useful tasks with the natural languages humans use. The input and output of an NLP system can be −

1) Speech
2) Written Text

Components of NLP

Natural Language Understanding (NLU)

Understanding involves the following tasks −

Mapping the given input in natural language into useful representations.

Analyzing different aspects of the language.

Natural Language Generation (NLG)

It is the process of producing meaningful phrases and sentences in the form of natural language from some internal representation.

It involves −

· **Text planning** − It includes retrieving the relevant content from knowledge base.

· **Sentence planning** − It includes choosing required words, forming meaningful phrases, setting tone of the sentence.

· **Text Realization** − It is mapping sentence plan into sentence structure.

The NLU is harder than NLG.

Difficulties in NLU

NL has an extremely rich form and structure.

It is very ambiguous. There can be different levels of ambiguity −

· **Lexical ambiguity** − It is at very primitive level such as word-level.

· For example, treating the word "board" as noun or verb?

· **Syntax Level ambiguity** − A sentence can be parsed in different ways.

· For example, "He lifted the beetle with red cap." − Did he use cap to lift the beetle or he lifted a beetle that had red cap?

· **Referential ambiguity** − Referring to something using pronouns. For example, Rima went to Gauri. She said, "I am tired." − Exactly who is tired?

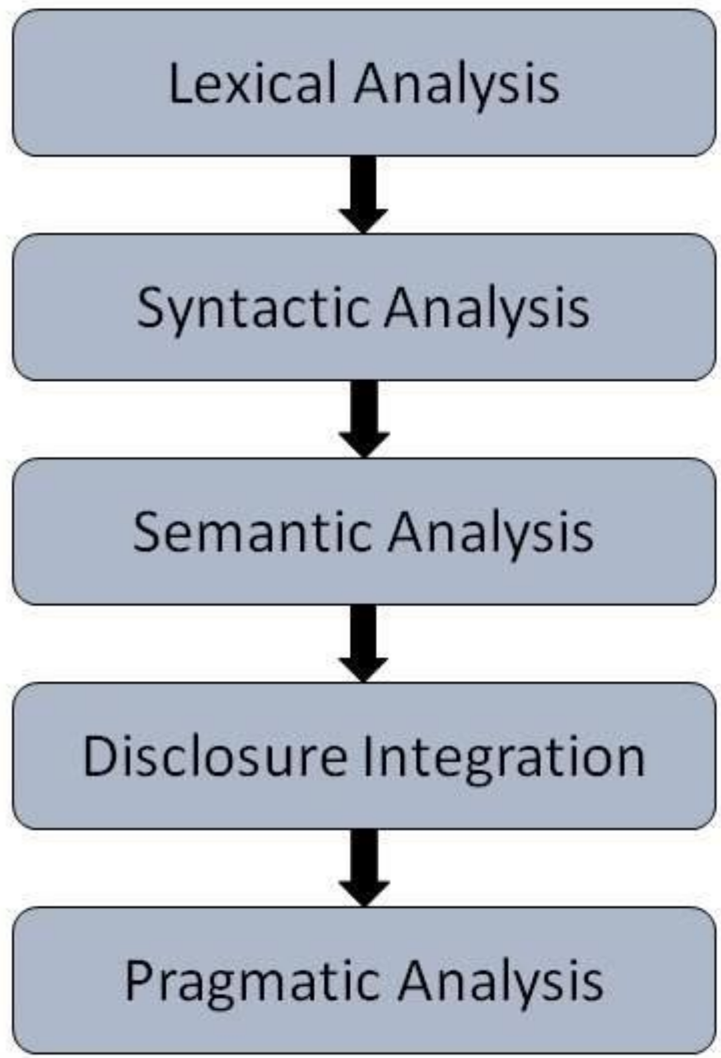
· One input can mean different meanings.

· Many inputs can mean the same thing.

NLP Terminology

- **Phonology** − It is study of organizing sound systematically.
- **Morphology** − It is a study of construction of words from primitive meaningful units.
- **Morpheme** − It is primitive unit of meaning in a language.
- **Syntax** − It refers to arranging words to make a sentence. It also involves determining the structural role of words in the sentence and in phrases.
- **Semantics** − It is concerned with the meaning of words and how to combine words into meaningful phrases and sentences.
- **Pragmatics** − It deals with using and understanding sentences in different situations and how the interpretation of the sentence is affected.
- **Discourse** − It deals with how the immediately preceding sentence can affect the interpretation of the next sentence.
- **World Knowledge** − It includes the general knowledge about the world.

Steps in NLP



Lexical Analysis
Syntactic Analysis
Semantic Analysis
Disclosure Integration
Pragmatic Analysis

There are general five steps −

· **Lexical Analysis** − It involves identifying and analysing the structure of words. Lexicon of a language means the collection of words and phrases in a language. Lexical analysis is dividing the whole chunk of txt into paragraphs, sentences, and words.

· **Syntactic Analysis (Parsing)** − It involves analysis of words in the sentence for grammar and arranging words in a manner that shows the relationship among the words. The sentence such as “The school goes to boy” is rejected by English syntactic analyser.

· **Semantic Analysis** − It draws the exact meaning or the dictionary meaning from the text. The text is checked for meaningfulness. It is done by mapping syntactic structures and objects in the task domain. The semantic analyser disregards sentence such as “hot ice-cream”.

· **Discourse Integration** − The meaning of any sentence depends upon the meaning of the sentence just before it. In addition, it also brings about the meaning of immediately succeeding sentence.

· **Pragmatic Analysis** − During this, what was said is re-interpreted on what it actually meant. It involves deriving those aspects of language which require real world knowledge.

AI - Expert Systems

What are Expert Systems?

The expert systems are the computer applications developed to solve complex problems in a particular domain, at the level of extra-ordinary human intelligence and expertise.

Characteristics of Expert Systems

High performance

Understandable

Reliable

Highly responsive

Components of Expert Systems

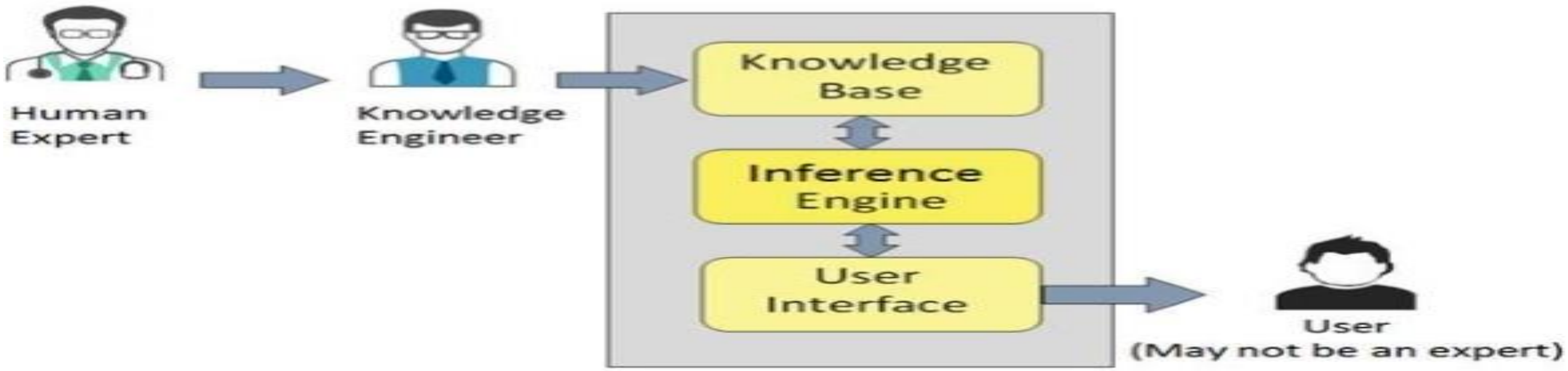
The components of ES include −

Knowledge Base

Inference Engine

User Interface

Let us see them one by one briefly −



Knowledge Base

It contains domain-specific and high-quality knowledge. Knowledge is required to exhibit intelligence. The success of any ES majorly depends upon the collection of highly accurate and precise knowledge.

What is Knowledge?

The data is collection of facts. The information is organized as data and facts about the task domain. **Data, information,** and **past experience** combined together are termed as knowledge.

Components of Knowledge Base

The knowledge base of an ES is a store of both, factual and heuristic knowledge.

- **Factual Knowledge** − It is the information widely accepted by the Knowledge Engineers and scholars in the task domain.
- **Heuristic Knowledge** − It is about practice, accurate judgement, one's ability of evaluation, and guessing.

Knowledge representation

It is the method used to organize and formalize the knowledge in the knowledge base. It is in the form of IF-THEN-ELSE rules.

Knowledge Acquisition

The success of any expert system majorly depends on the quality, completeness, and accuracy of the information stored in the knowledge base.

The knowledge base is formed by readings from various experts, scholars, and the **Knowledge Engineers**. The knowledge engineer is a person with the qualities of empathy, quick learning, and case analyzing skills.

He acquires information from subject expert by recording, interviewing, and observing him at work, etc. He then categorizes and organizes the information in a meaningful way, in the form of IF-THEN-ELSE rules, to be used by interference machine. The knowledge engineer also monitors the development of the ES.

Inference Engine

Use of efficient procedures and rules by the Inference Engine is essential in deducting a correct, flawless solution.

In case of knowledge-based ES, the Inference Engine acquires and manipulates the knowledge from the knowledge base to arrive at a particular solution.

In case of rule based ES, it −

- Applies rules repeatedly to the facts, which are obtained from earlier rule application.
- Adds new knowledge into the knowledge base if required.
- Resolves rules conflict when multiple rules are applicable to a particular case.

To recommend a solution, the Inference Engine uses the following strategies −

Forward Chaining

Backward Chaining

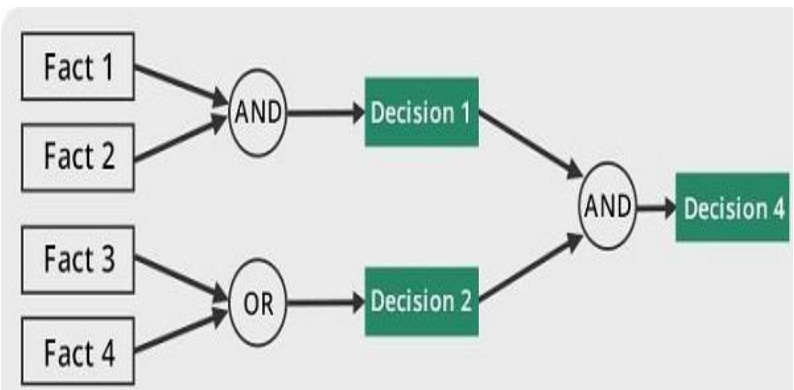
Forward Chaining

It is a strategy of an expert system to answer the question, **“What can happen next?”**

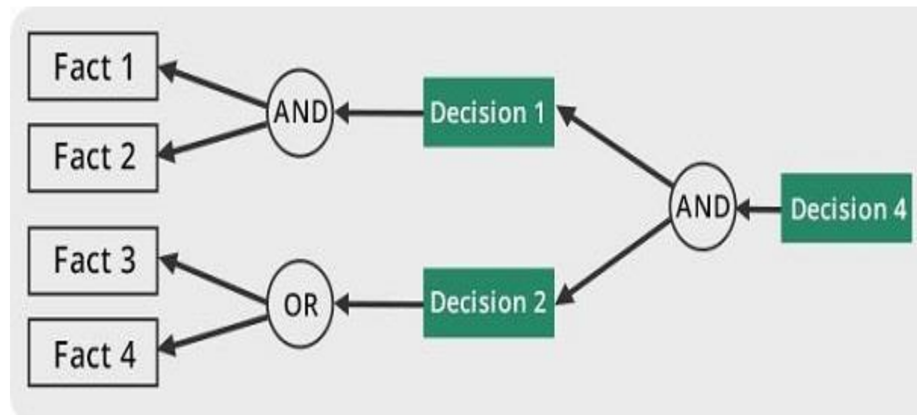
Here, the Inference Engine follows the chain of conditions and derivations and finally deduces the outcome. It considers all the facts and rules, and sorts them before concluding to a solution.

This strategy is followed for working on conclusion, result, or effect. For example, prediction of share market status as an effect of changes in interest rates.

FC



BC



Backward Chaining

With this strategy, an expert system finds out the answer to the question, **“Why this happened?”**

On the basis of what has already happened, the Inference Engine tries to find out which conditions could have happened in the past for this result. This strategy is followed for finding out cause or reason. For example, diagnosis of blood cancer in humans.

Applications of Expert System

Application	Description
Design Domain	Camera lens design, automobile design.
Medical Domain	Diagnosis Systems to deduce cause of disease from observed data, conduction medical operations on humans.
Monitoring Systems	Comparing data continuously with observed system or with prescribed behaviour such as leakage monitoring in long petroleum pipeline.
Process Control Systems	Controlling a physical process based on monitoring.
Knowledge Domain	Finding out faults in vehicles, computers.
Finance/Commerce	Detection of possible fraud, suspicious transactions, stock market trading, Airline scheduling, cargo scheduling.

Capabilities of Expert Systems

The expert systems are capable of –

1 Advising
2 Instructing and assisting human in decision making
3 Demonstrating
4 Deriving a solution
5 Diagnosing
6 Explaining
7 Interpreting input
8 Predicting results
9 justifying the conclusion
10 Suggesting alternative options to a problem

They are incapable of –

1 Substituting human decision makers
2 Possessing human capabilities
3 Producing accurate output for inadequate knowledge base
4 Refining their own knowledge
5 Components of Expert Systems
6 The components of ES include –
7 Knowledge Base
8 Inference Engine
9 User Interface

Artificial Intelligence - Robotics

Robotics is a domain in artificial intelligence that deals with the study of creating intelligent and efficient robots.

What are Robots?

Robots are the artificial agents acting in real world environment.

Objective

Robots are aimed at manipulating the objects by perceiving, picking, moving, modifying the physical properties of object, destroying it, or to have an effect thereby freeing manpower from doing repetitive functions without getting bored, distracted, or exhausted.

What is Robotics?

Robotics is a branch of AI, which is composed of Electrical Engineering, Mechanical Engineering, and Computer Science for designing, construction, and application of robots.

Aspects of Robotics

- The robots have **mechanical construction**, form, or shape designed to accomplish a particular task.
- They have **electrical components** which power and control the machinery.
- They contain some level of **computer program** that determines what, when and how a robot does something.

Difference in Robot System and Other AI Program

AI Programs	Robots
They usually operate in computer-stimulated worlds.	They operate in real physical world
The input to an AI program is in symbols and rules.	Inputs to robots is analog signal in the form of speech waveform or images
They need general purpose computers to operate on.	They need special hardware with sensors and effectors.

Robot Locomotion

Locomotion is the mechanism that makes a robot capable of moving in its environment.

There are various types of locomotions –

1 Legged
2 Wheeled
3 Combination of Legged and Wheeled Locomotion
4 Tracked slip/skid

Legged Locomotion

- This type of locomotion consumes more power while demonstrating walk, jump, trot, hop, climb up or down, etc.
- It requires more number of motors to accomplish a movement. It is suited for rough as well as smooth terrain where irregular or too smooth surface makes it consume more power for a wheeled locomotion. It is little difficult to implement because of stability issues.
- It comes with the variety of one, two, four, and six legs. If a robot has multiple legs then leg coordination is necessary for locomotion.

The total number of possible **gaits** (a periodic sequence of lift and release events for each of the total legs) a robot can travel depends upon the number of its legs.

If a robot has k legs, then the number of possible events N = (2k-1)!.

In case of a two-legged robot (k=2), the number of possible events is N = (2k-1)! = (2*2-1)! = 3! = 6.

Hence there are six possible different events −

Lifting the Left leg

Releasing the Left leg

Lifting the Right leg

Releasing the Right leg

Lifting both the legs together

Releasing both the legs together

In case of k=6 legs, there are 39916800 possible events. Hence the complexity of robots is directly proportional to the number of legs.

Wheeled Locomotion

It requires fewer number of motors to accomplish a movement. It is little easy to implement as there are less stability issues in case of more number of wheels. It is power efficient as compared to legged locomotion.

- **Standard wheel** − Rotates around the wheel axle and around the contact
- **Castor wheel** − Rotates around the wheel axle and the offset steering joint.
- **Swedish 45° and Swedish 90° wheels** − Omni-wheel, rotates around the contact point, around the wheel axle, and around the rollers.
- **Ball or spherical wheel** − Omnidirectional wheel, technically difficult to implement.

Slip/Skid Locomotion

In this type, the vehicles use tracks as in a tank. The robot is steered by moving the tracks with different speeds in the same or opposite direction. It offers stability because of large contact area of track and ground.

Components of a Robot

Robots are constructed with the following −

- **Power Supply** − The robots are powered by batteries, solar power, hydraulic, or pneumatic power sources.
- **Actuators** − They convert energy into movement.
- **Electric motors (AC/DC)** − They are required for rotational movement.
- **Pneumatic Air Muscles** − They contract almost 40% when air is sucked in them.
- **Muscle Wires** − They contract by 5% when electric current is passed through them.
- **Piezo Motors and Ultrasonic Motors** − Best for industrial robots.
- **Sensors** − They provide knowledge of real time information on the task environment. Robots are equipped with vision sensors to be to compute the depth in the environment. A tactile sensor imitates the mechanical properties of touch receptors of human fingertips.

Computer Vision

This is a technology of AI with which the robots can see. The computer vision plays vital role in the domains of safety, security, health, access, and entertainment.

Computer vision automatically extracts, analyzes, and comprehends useful information from a single image or an array of images. This process involves development of algorithms to accomplish automatic visual comprehension.

Hardware of Computer Vision System

This involves −

Power supply

Image acquisition device such as camera

a processor

a software

A display device for monitoring the system

Accessories such as camera stands, cables, and connectors

Tasks of Computer Vision

· **OCR** − In the domain of computers, Optical Character Reader, a software to convert scanned documents into editable text, which accompanies a scanner.

· **Face Detection** − Many state-of-the-art cameras come with this feature, which enables to read the face and take the picture of that perfect expression. It is used to let a user access the software on correct match.

· **Object Recognition** − They are installed in supermarkets, cameras, high-end cars such as BMW, GM, and Volvo.

· **Estimating Position** − It is estimating position of an object with respect to camera as in position of tumor in human's body.

Application Domains of Computer Vision

Agriculture
Autonomous vehicles
Biometrics
Character recognition
Forensics, security, and surveillance
Industrial quality inspection
Face recognition
Gesture analysis
Geoscience
Medical imagery
Pollution monitoring
Process control
Remote sensing
Robotics
Transport

AI - Neural Networks

Yet another research area in AI, neural networks, is inspired from the natural neural network of human nervous system.

What are Artificial Neural Networks (ANNs)?

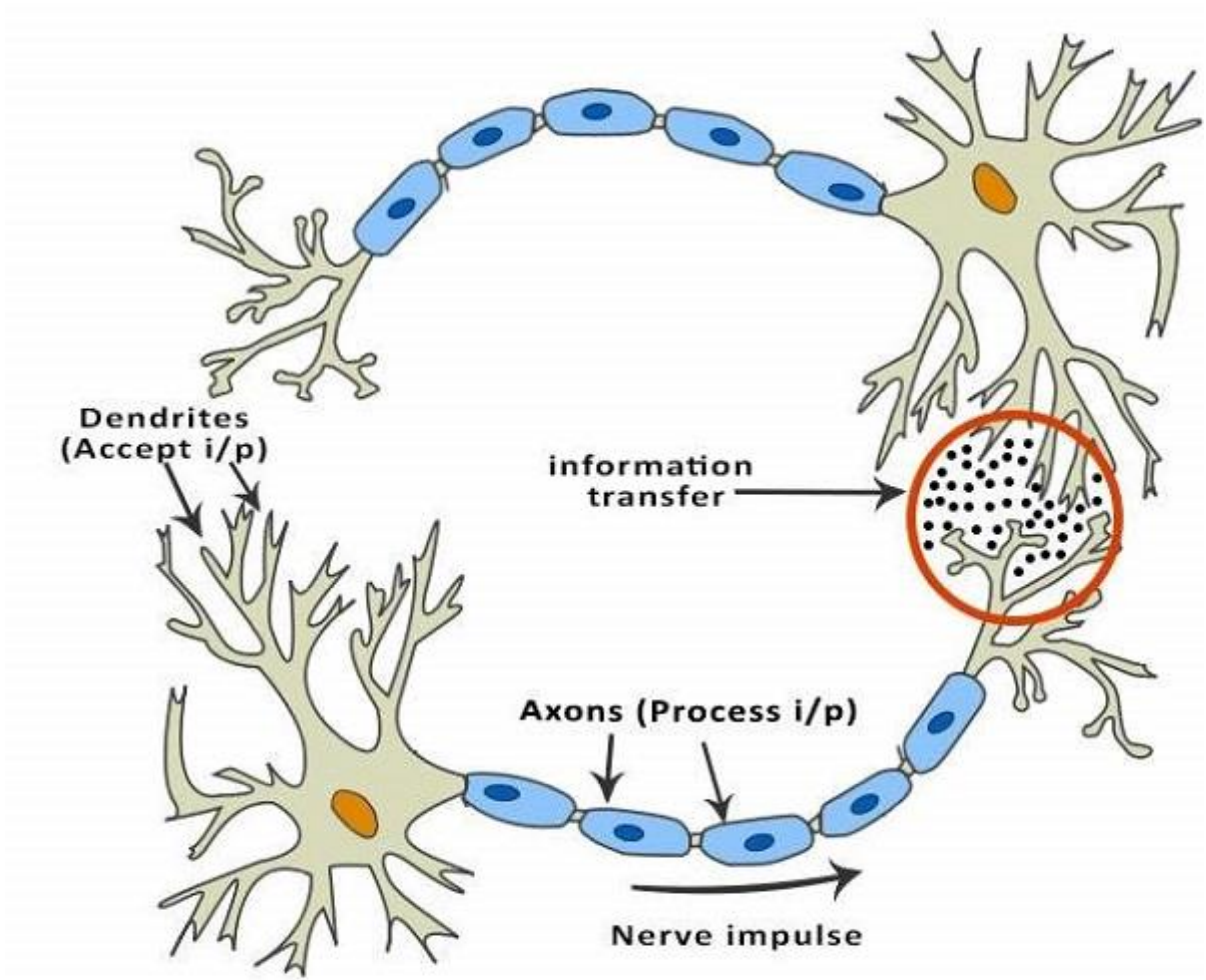
The inventor of the first neuro computer, Dr. Robert Hecht-Nielsen, defines a neural network as −

"...a computing system made up of a number of simple, highly interconnected processing elements, which process information by their dynamic state response to external inputs."

Basic Structure of ANNs

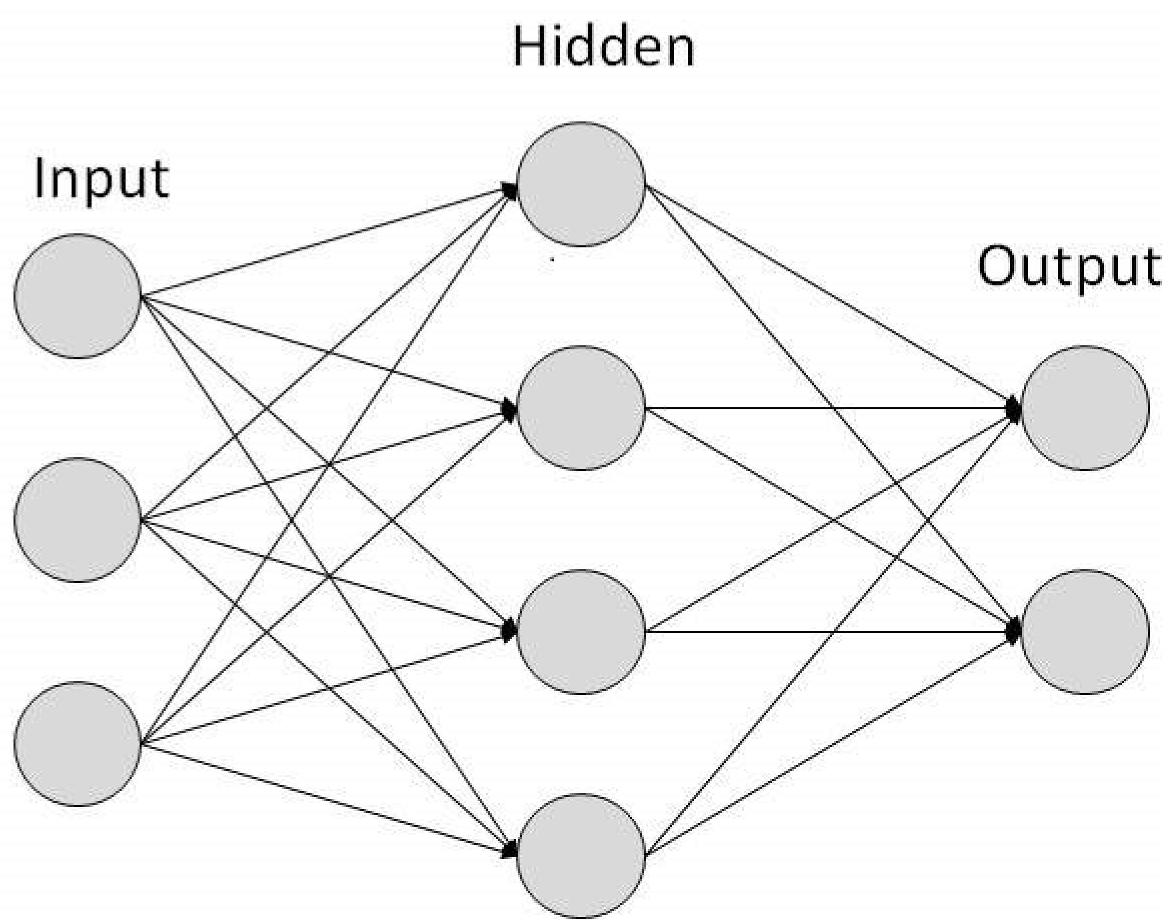
The idea of ANNs is based on the belief that working of human brain by making the right connections, can be imitated using silicon and wires as living **neurons** and **dendrites**.

The human brain is composed of 86 billion nerve cells called **neurons.** They are connected to other thousand cells by **Axons.** Stimuli from external environment or inputs from sensory organs are accepted by dendrites. These inputs create electric impulses, which quickly travel through the neural network. A neuron can then send the message to other neuron to handle the issue or does not send it forward.



Dendrites
(Accept i/p)
information
transfer
Axons (Process i/p)
Nerve impulse

ANNs are composed of multiple **nodes**, which imitate biological **neurons** of human brain. The neurons are connected by links and they interact with each other. The nodes can take input data and perform simple operations on the data. The result of these operations is passed to other neurons. The output at each node is called its **activation** or **node value.**
Each link is associated with **weight.** ANNs are capable of learning, which takes place by altering weight values. The following illustration shows a simple ANN −



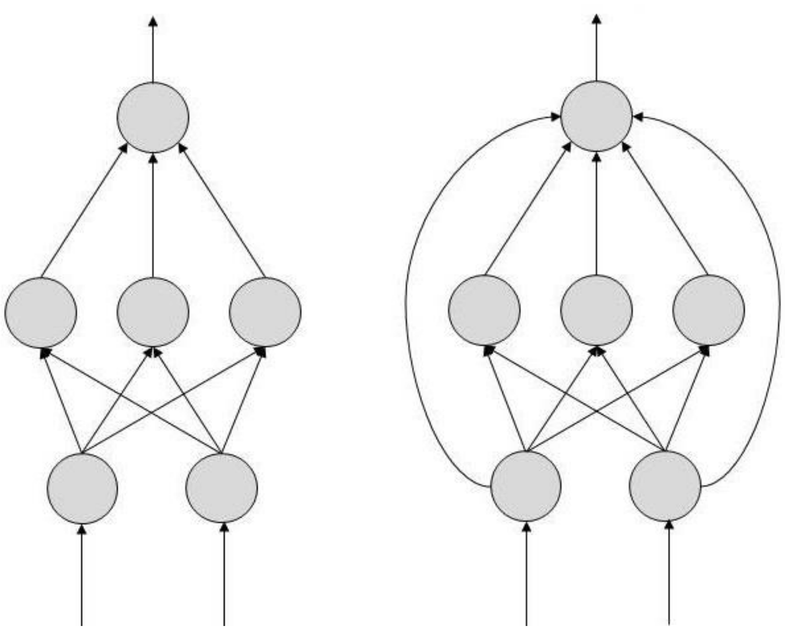
Types of Artificial Neural Networks

There are two Artificial Neural Network topologies − **FeedForward** and **Feedback.**

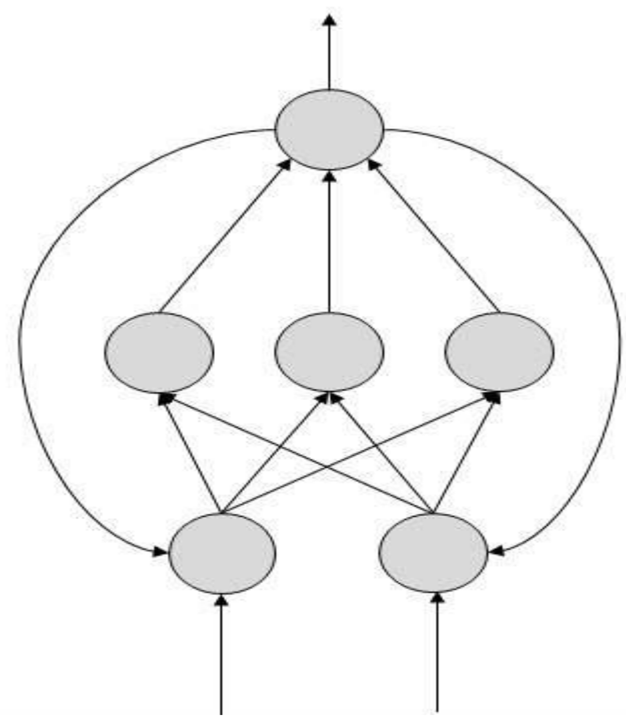
FeedForward ANN

The information flow is unidirectional. A unit sends information to other unit from which it does not receive any information. There are no feedback loops. They are used in pattern generation/recognition/classification. They have fixed inputs and outputs.\

FF



FB



Working of ANNs

In the topology diagrams shown, each arrow represents a connection between two neurons and indicates the pathway for the flow of information. Each connection has a weight, an integer number that controls the signal between the two neurons.

If the network generates a “good or desired” output, there is no need to adjust the weights. However, if the network generates a “poor or undesired” output or an error, then the system alters the weights in order to improve subsequent results.

Machine Learning in ANNs

ANNs are capable of learning and they need to be trained. There are several learning strategies −

·

Supervised Learning − It involves a teacher that is scholar than the ANN itself. For example, the teacher feeds some example data about which the teacher already knows the answers. For example, pattern recognizing. The ANN comes up with guesses while recognizing. Then the teacher provides the ANN with the answers. The network then compares it guesses with the teacher’s “correct” answers and makes adjustments according to errors.

· **Unsupervised Learning** − It is required when there is no example data set with known answers. For example, searching for a hidden pattern. In this case, clustering i.e. dividing a set of elements into groups according to some unknown pattern is carried out based on the existing data sets present.

· **Reinforcement Learning** − This strategy built on observation. The ANN makes a decision by observing its environment. If the observation is negative, the network adjusts its weights to be able to make a different required decision the next time.

Back Propagation Algorithm

It is the training or learning algorithm. It learns by example. If you submit to the algorithm the example of what you want the network to do, it changes the network's weights so that it can produce desired output for a particular input on finishing the training.

Back Propagation networks are ideal for simple Pattern Recognition and Mapping Tasks.

Bayesian Networks (BN)

These are the graphical structures used to represent the probabilistic relationship among a set of random variables. Bayesian networks are also called **Belief Networks** or **Bayes Nets.** BNs reason about uncertain domain.

In these networks, each node represents a random variable with specific propositions. For example, in a medical diagnosis domain, the node Cancer represents the proposition that a patient has cancer.

The edges connecting the nodes represent probabilistic dependencies among those random variables. If out of two nodes, one is affecting the other then they must be directly connected in the directions of the effect. The strength of the relationship between variables is quantified by the probability associated with each node.

There is an only constraint on the arcs in a BN that you cannot return to a node simply by following directed arcs. Hence the BNs are called Directed Acyclic Graphs (DAGs).

BNs are capable of handling multivalued variables simultaneously. The BN variables are composed of two dimensions –

Range of prepositions

Building a Bayesian Network

A knowledge engineer can build a Bayesian network. There are a number of steps the knowledge engineer needs to take while building it.

Example problem − *Lung cancer.* A patient has been suffering from breathlessness. He visits the doctor, suspecting he has lung cancer. The doctor knows that barring lung cancer, there are various other possible diseases the patient might have such as tuberculosis and bronchitis.

Gather Relevant Information of Problem

Is the patient a smoker? If yes, then high chances of cancer and bronchitis.

Is the patient exposed to air pollution? If yes, what sort of air pollution?

Take an X-Ray positive X-ray would indicate either TB or lung cancer.

Identify Interesting Variables

The knowledge engineer tries to answer the questions −

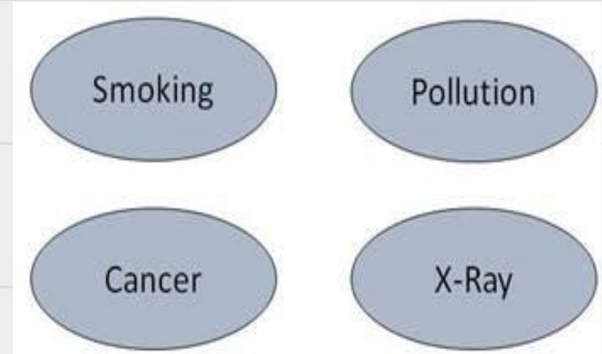
Which nodes to represent?

What values can they take? In which state can they be?

For now let us consider nodes, with only discrete values. The variable must take on exactly one of these values at a time.

Common types of discrete nodes are −

- **Boolean nodes** − They represent propositions, taking binary values TRUE (T) and FALSE (F).
- **Ordered values** − A node *Pollution* might represent and take values from {low, medium, high} describing degree of a patient’s exposure to pollution.
- **Integral values** − A node called *Age* might represent patient’s age with possible values from 1 to 120. Even at this early stage, modeling choices are being made.

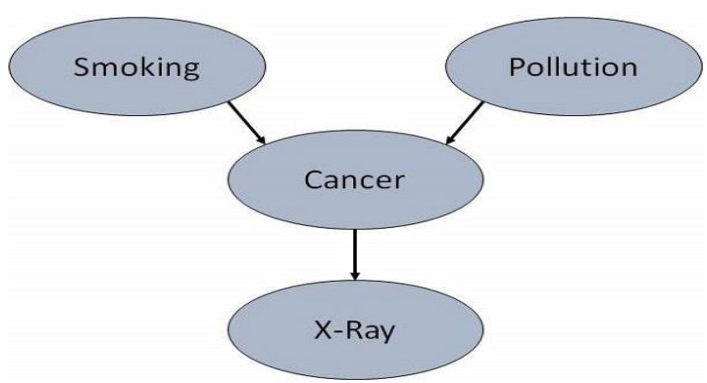
node Name	Type	Value	Nodes Creation
Polution	Binary	{LOW, HIGH, MEDIUM}	
Smoker	Boolean	{TRUE, FASLE}	
Lung-Cancer	Boolean	{TRUE, FASLE}	
X-Ray	Binary	{Positive, Negative}	

Create Arcs between Nodes

Topology of the network should capture qualitative relationships between variables.

For example, what causes a patient to have lung cancer? - Pollution and smoking. Then add arcs from node *Pollution* and node *Smoker* to node *Lung-Cancer.*

Similarly if patient has lung cancer, then X-ray result will be positive. Then add arcs from node *Lung-Cancer* to node *X-Ray.*



Specify Topology

Conventionally, BNs are laid out so that the arcs point from top to bottom. The set of parent nodes of a node X is given by Parents(X).

The *Lung-Cancer* node has two parents (reasons or causes): *Pollution* and *Smoker*, while node *Smoker* is an **ancestor** of node *X-Ray*. Similarly, *X-Ray* is a child (consequence or effects) of node *Lung-Cancer* and **successor** of nodes *Smoker* and *Pollution.*

Conditional Probabilities

Now quantify the relationships between connected nodes: this is done by specifying a conditional probability distribution for each node. As only discrete variables are considered here, this takes the form of a **Conditional Probability Table (CPT).**

First, for each node we need to look at all the possible combinations of values of those parent nodes. Each such combination is called an **instantiation** of the parent set. For each distinct instantiation of parent node values, we need to specify the probability that the child will take.

For example, the *Lung-Cancer* node's parents are *Pollution* and *Smoking.* They take the possible values = { (H,T), (H,F), (L,T), (L,F)}. The CPT specifies the probability of cancer for each of these cases as <0.05, 0.02, 0.03, 0.001> respectively.

Each node will have conditional probability associated as follows −

Smoking
P(S = T)
0.30

Pollution
P(P = L)
0.90

Lung-Cancer		
P	**S**	**P (C = T|P, S)**
H	T	0.05
H	F	0.02
L	T	0.03
L	F	0.001

X-Ray	
C	**X = (Pos | C)**
T	0.90
F	0.20

Applications of Neural Networks

They can perform tasks that are easy for a human but difficult for a machine −

- **Aerospace** − Autopilot aircrafts, aircraft fault detection.
- **Automotive** − Automobile guidance systems.
- **Military** − Weapon orientation and steering, target tracking, object discrimination, facial recognition, signal/image identification.
- **Electronics** − Code sequence prediction, IC chip layout, chip failure analysis, machine vision, voice synthesis.
- **Financial** − Real estate appraisal, loan advisor, mortgage screening, corporate bond rating, portfolio trading program, corporate financial analysis, currency value prediction, document readers, credit application evaluators.
- **Industrial** − Manufacturing process control, product design and analysis, quality inspection systems, welding quality analysis, paper quality prediction, chemical product design analysis, dynamic modeling of chemical process systems, machine maintenance analysis, project bidding, planning, and management.
- **Medical** − Cancer cell analysis, EEG and ECG analysis, prosthetic design, transplant time optimizer.
-

Speech − Speech recognition, speech classification, text to speech conversion.

· **Telecommunications** − Image and data compression, automated information services, real-time spoken language translation.

· **Transportation** − Truck Brake system diagnosis, vehicle scheduling, routing systems.

· **Software** − Pattern Recognition in facial recognition, optical character recognition, etc.

· **Time Series Prediction** − ANNs are used to make predictions on stocks and natural calamities.

· **Signal Processing** − Neural networks can be trained to process an audio signal and filter it appropriately in the hearing aids.

· **Control** − ANNs are often used to make steering decisions of physical vehicles.

· **Anomaly Detection** − As ANNs are expert at recognizing patterns, they can also be trained to generate an output when something unusual occurs that misfits the pattern.

Artificial Intelligence - Issues

AI is developing with such an incredible speed, sometimes it seems magical. There is an opinion among researchers and developers that AI could grow so immensely strong that it would be difficult for humans to control.

Humans developed AI systems by introducing into them every possible intelligence they could, for which the humans themselves now seem threatened.

Threat to Privacy

An AI program that recognizes speech and understands natural language is theoretically capable of understanding each conversation on e-mails and telephones.

Threat to Human Dignity

AI systems have already started replacing the human beings in few industries. It should not replace people in the sectors where they are holding dignified positions which are pertaining to ethics such as nursing, surgeon, judge, police officer, etc.

Threat to Safety

The self-improving AI systems can become so mighty than humans that could be very difficult to stop from achieving their goals, which may lead to unintended consequences.

Thank You