1 UNIT

Introduction

CONTENTS

PART-1

Introduction-definition.

Questions-Answers

Long Answer Type and Medium Answer Type Questions

Que 1.1. What do you mean by artificial intelligence ? Define its goals.

Answer

1. Artificial Intelligence (AI) is an area of computer science that emphasizes the creation of intelligent machines that work and reacts like humans.
2. AI has become an essential part of the technology industry.
3. Research associated with artificial intelligence is highly technical and specialized. The core problems of artificial intelligence include programming computers for certain traits such as :
   i. Knowledge
   ii. Reasoning
   iii. Problem solving
   iv. Perception
   v. Learning
   vi. Planning
   vii. Ability to manipulate and move objects

Goals of AI :

1. **To create expert systems :** The systems which exhibit intelligent behaviour, learn, demonstrate, explain, and advice its users.
2. **To implement human intelligence in machines :** Creating systems that understand, think, learn, and behave like humans.

Que 1.2. What are the different branches of artificial intelligence ? Discuss some of the branches and progress made in their fields.

Answer

Different branches of AI and progress made in these fields :

1. **Machine Learning (ML) :**
   i. ML is a method where the target is defined and the steps to reach that target are learned by the machine itself by training.

ii. For example, to identify a simple object such as an apple or orange.

The target is achieved by showing multiple pictures of object and thereby allowing the machine to define the steps to identify it like an apple or an orange.

2. **Natural Language Processing (NLP) :**
   i. NLP is defined as the automatic manipulation of natural language, like speech and text, by software.
   ii. For example, e-mail spam detection which has improved the mail system.
3. **Vision :** Machine vision captures and analyses visual information using a camera, analog-to-digital conversion, and digital signal processing.
4. **Robotics :**
   i. Robotics is a field of engineering focused on the design and manufacturing of robots.
   ii. Robots are used where the tasks are difficult for humans to perform.
   iii. For example, car assembly lines, in hospitals, office cleaner, serving foods and preparing foods in hotels etc.

Que 1.3. Define the role of the machine intelligence in the human life. **AKTU 2017-18, Marks 10**

Answer

1. Machine intelligence is the intelligence provided to the particular machine to achieve the goals of the problems in AI.
2. It is defined as the embedding of intelligence in the machine so that the machine can behave like a human.
3. In human life, machine learning solves many problems of daily purpose of the human.
4. There are many problems which require intelligence such as complex arithmetic which is done by machine very easily.
5. Machine learning plays an important role in following areas :
   i. Learning :
      a. Learning means to acquire new things from the set of given knowledge or experiences.
      b. It refers to the change in subject's behaviour to a given situation brought by repeated experiences in that situation.
   ii. Reasoning :
      a. Reasoning means to infer facts from given facts.
      b. Inferences are classified as either deductive or inductive and the reasoning is to draw inferences appropriate to the situation.

iii. Problem solving :

a. To solve problem means to move towards the goal.

b. In this, set of rules are defined and a goal is also defined which is to be achieved by using these rules.

iv. Language understanding :

a. It means to understand natural language meaning.

b. A language is a system of signs having meaning-by-convention.

c. The meaning-by-convention is distinctive of language and is very different from natural meaning.

Que 1.4. Describe the four categories under which AI is classified with examples. **AKTU 2020-21, Marks 07**

Answer

Distinct categories of AI namely :

1. **Reactive machines :** These are the most basic type of AI and are purely reactive. They neither can form memories nor can use past experiences to form decisions.

   For example : IBM's Deep Blue chess-playing is supercomputer which choose most optimal of the chess moves and beat the opponent. Apart from a rarely used chess-specific rule against repeating the same move three times, Deep Blue ignores everything before the present moment, thus not storing any memories. This type of AI just perceives the world, the chess game in the case of Deep Blue, and acts on it.

2. **Limited memory :** These machines can look into the past. Not the ability to predict what happened in the past, but the usage of memories to form decisions.

   For example : Self-driving cars. They observe other cars' speed and directions and act accordingly. This requires monitoring of how a car is driven for a specific amount of time. Just like how humans observe and learn the specifics. These pieces of information are not stored in the library of experiences of the machines, unlike humans. We humans automatically save everything in the library of our experiences and can learn from it, but limited memory machines can't.

3. **Theory of mind :** These are types of machines that can understand that people have beliefs, emotions, expectations, etc. and have some of their own.

   A "theory of mind" machine can think emotionally and can respond with emotions.

   For example : AI like Sophia, the research is not complete yet, these machines have a notion of not just the world, but also the existing entities of the world, like human beings, animals, etc.

4. **Self-Awareness :** These types of machines can be called human equivalents.

   For example : The difference between "theory of mind" and "self-awareness" AI. The feeling of I want to play is different from the feeling of I know I want to play. In the latter, there is a sense of consciousness and is a characteristic of a self-aware machine, while the former feeling is a characteristic of a theory-of-mind machine. Self-aware machines will have the ability to predict others' feelings.

PART-2

Future of Artificial Intelligence.

Questions-Answers

Long Answer Type and Medium Answer Type Questions

Que 1.5. Describe briefly the evolution of artificial intelligence.

AKTU 2018-19, Marks 10

Answer

Evolution of AI :

1. **Beginning of AI (1943) :** The concept of AI began around 1943. AI is not limited to the computer sciences disciplines, but can be seen in various other areas.
2. **AI knowledge-based expert system (1970) :** An AI system often uses a rule-based system to capture knowledge in the form of if-then statements or as decision trees.
3. **Machine learning (1998) :** There are two types of machine learning :
   i. **Formal :** The formal type of machine learning is a computer program that learns from experience in respect to some task and increases performance based on that experience.
   ii. **Informal :** The informal involves giving computers the ability to learn without explicitly programming the capability.
4. **Supervised learning (2004) :** The supervised learning is based on giving the correct answers and having the computer mapping inputs to outputs. For example,
   i. **Spam filters :** Software is trained to learn and distinguish between spam and non-spam messages (For example, e-mail filters).
   ii. **Facial recognition :** It is used by cameras to focus and via photo editing software to tag persons (For example, Facebook).

5. **Unsupervised learning (2010) :** Unsupervised learning is the reverse of supervised learning where the correct answers are unknown. For example,
   i. **Clustering algorithm :** Used to find patterns in datasets and then group that data into different coherent clusters.
   ii. **Market segmentation :** Targeting customers based on regions, likes, dislikes, when the consumer makes purchases, etc. This is considered targeted marketing.
6. **Genetic programming (2010) :** Genetic programming is an idea that uses evolutionary process to improve algorithms.
7. **Future of AI (2019 onwards) :** There are many challenges in mimicking human intelligence. Humans acquire common senses that are intuitive but hard to reason rationally. For example, the colour of a blue car is blue.

PART-3

Characteristics of Intelligent Agents, Typical Intelligent Agents.

Questions-Answers

Long Answer Type and Medium Answer Type Questions

Que 1.6. What do you mean by agent and agent program ? How do you ensure that an agent program is an intelligent agent program ? **AKTU 2014-15, Marks 05**

Answer

1. An agent can be anything that perceive its environment through sensors and act upon that environment through actuators. An agent runs in the cycle of perceiving, thinking, and acting.
2. To understand the structure of intelligent agents, we should be familiar with architecture and agent program.

   Agent = Architecture + Agent Program
3. Architecture is the machinery that the agent executes on. It is a device with sensors and actuators, for example : a robotic car, a camera, a PC.
4. Agent program is an implementation of an agent function.
5. An agent function is a map from the percept sequence to an action.
6. An agent program is an intelligent agent program if it follows the weak notion (*i.e.*, flexibility, interactivity and autonomy) and strong notion (*i.e.*, information-related states, connotative states and affective states).

Que 1.7. **What is intelligent agent ? Describe basic kinds of agent programs.** **AKTU 2015-16, Marks 05**

OR

Define intelligent agent. Explain various types of agent program with suitable example. **AKTU 2019-20, Marks 07**

Answer

Intelligent agent : An intelligent agent is an autonomous entity which acts upon an environment using sensors and actuators for achieving goals. An intelligent agent may learn from the environment to achieve their goals.

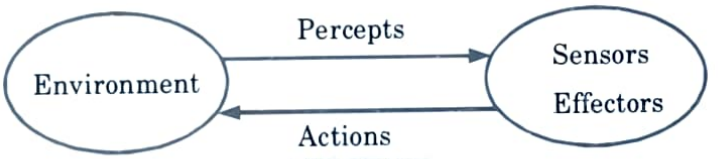


Fig. 1.7.1.

Basic kinds of agent programs are :

1. **Simple reflex agent :**
   i. The simple reflex agents are the simplest agents. These agents take decisions on the basis of the current percepts and ignore the rest of the percept history.
   ii. These agents only succeed in the fully observable environment.
   iii. The simple reflex agent works on condition-action rule, which means it maps the current state to action.

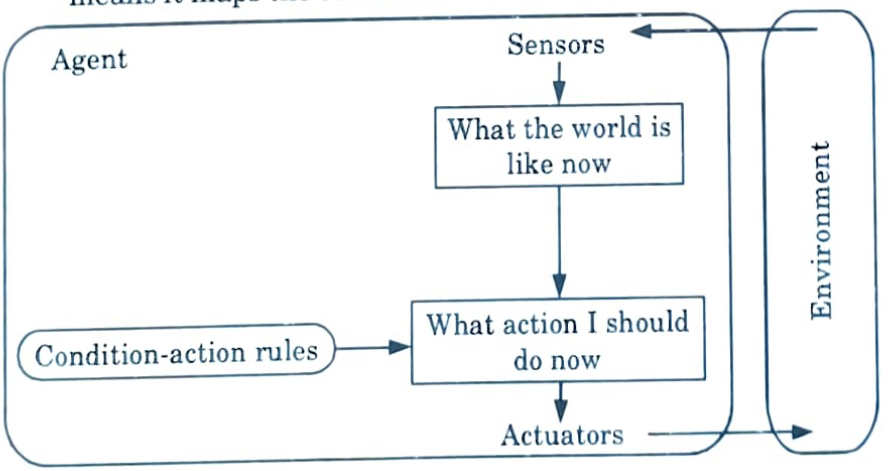


Fig. 1.7.2. Simple reflex agent.

   iv. It acts according to a rule whose condition matches the current state, as defined by the percept.

2. **Model-based reflex agent :**
   i. The model-based agent can work in a partially observable environment, and track the situation.

ii. A model-based agent has two important factors :

a. **Model :** It is knowledge about "how things happen in the world," so it is called model-based agent.

b. **Internal state :** It is a representation of the current state based on percept history.

iii. These agents have the model, "which is knowledge of the world" and based on the model they perform actions.

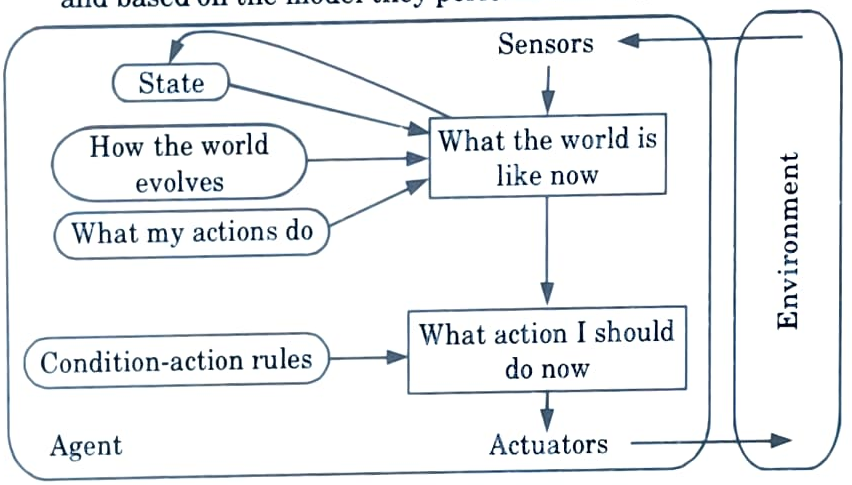


Fig. 1.7.3. Model-based agent.

iv. It keeps track of the current state of the world using an internal model and then chooses an action.

3. **Goal-based agents :**

i. The knowledge of the current state environment is not always sufficient to decide for an agent to what to do.

ii. The agent needs to know its goal which describes desirable situations. They choose an action, so that they can achieve the goal.

iii. These agents may have to consider a long sequence of possible actions before deciding whether the goal is achieved or not.

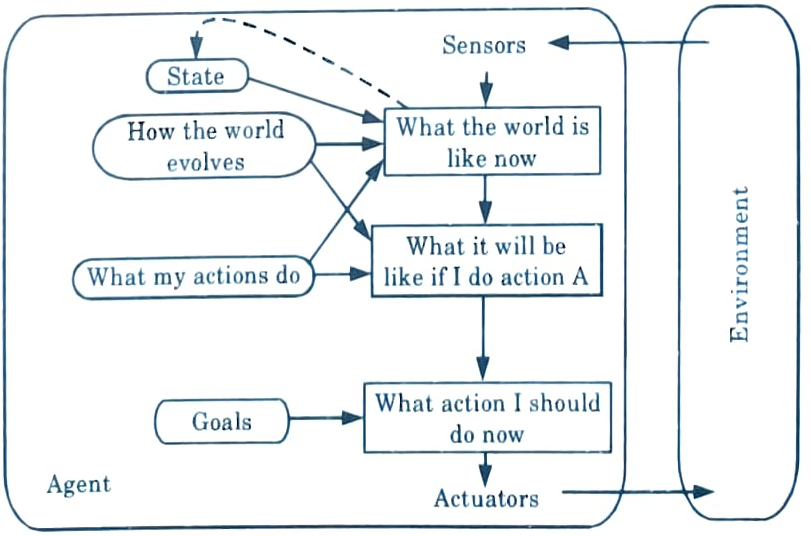


Fig. 1.7.4. Goal-based agent.

iv. It keeps track of the world state as well as a set of goals it is trying to achieve, and chooses an action that will (eventually) lead to the achievement of its goals.

4. Utility-based agents :

i. Utility-based agent is the best way to achieve the goal.

ii. The utility-based agent is useful when there are multiple possible alternatives, and an agent has to choose in order to perform the best action.

iii. The utility function maps each state to a real number to check how efficiently each action achieves the goals.

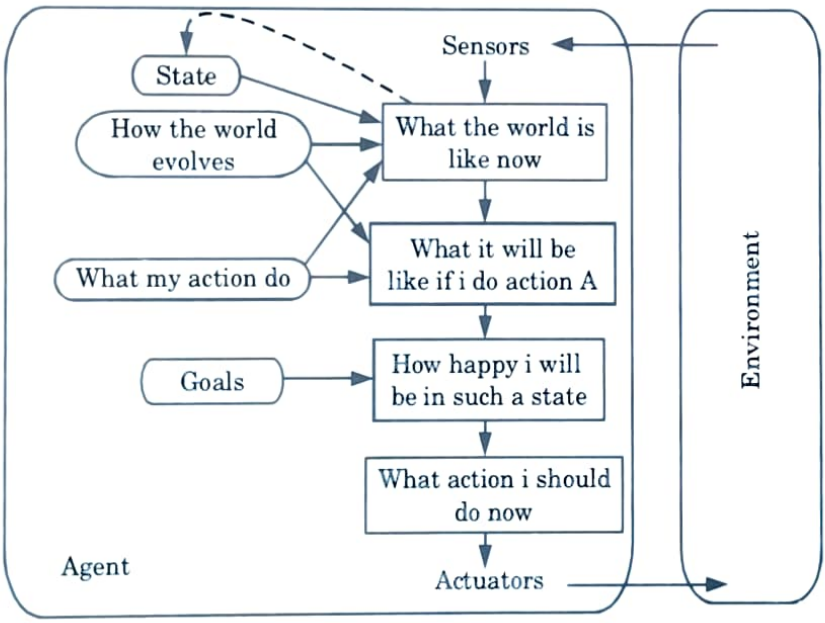


Fig. 1.7.5. Utility-based agent.

iv. It uses a model of the world, along with a utility function that measures its preferences among states of the world. Then it chooses the action that leads to the best expected utility, where expected utility.

Que 1.8. Describe the characteristics of intelligent system ?

Answer

Characteristics of intelligent agent (IA) :

1. The IA must learn and improve through interaction with the environment.
2. The IA must adapt online and in the real time situation.
3. The IA must learn quickly from large amounts of data.

4. The IA must accommodate new problem solving rules incrementally.
5. The IA must have memory which must exhibit storage and retrieval capacities.
6. The IA should be able to analyze self in terms of behaviour, error and success.

Que 1.9. State the various properties of environment.

AKTU 2018-19, Marks 10

Answer

Properties of environments :

1. **Discrete / Continuous :** If there are a limited number of distinct, clearly defined, states of the environment, the environment is discrete otherwise it is continuous.
2. **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.
3. **Dynamic / Static :** If the environment is changing for agents action then the environment is dynamic for that agent, otherwise it is static.
4. **Single agent / Multiple agents :** An agent operating by itself in an environment is single agent. Multiple agent is when other agents are present. Other agent is anything that changes from step to step.
5. **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; otherwise it is inaccessible.
6. **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.
7. **Episodic / Non-episodic :** If the agent's experience is divided into atomic episodes and in each episodes the agent receives a percept and then performs a single action, then the environment is episodic; otherwise it is non-episodic.

Que 1.10. Explain learning agent with its architecture.

Answer

1. A learning agent is a tool in AI that is capable of learning from its experiences.
2. Learning agents are able to perform tasks, analyze performance and look for new ways to improve on those tasks.

Architecture of learning agent :

1. A learning agent can be divided into four components as shown in the Fig. 1.10.1.

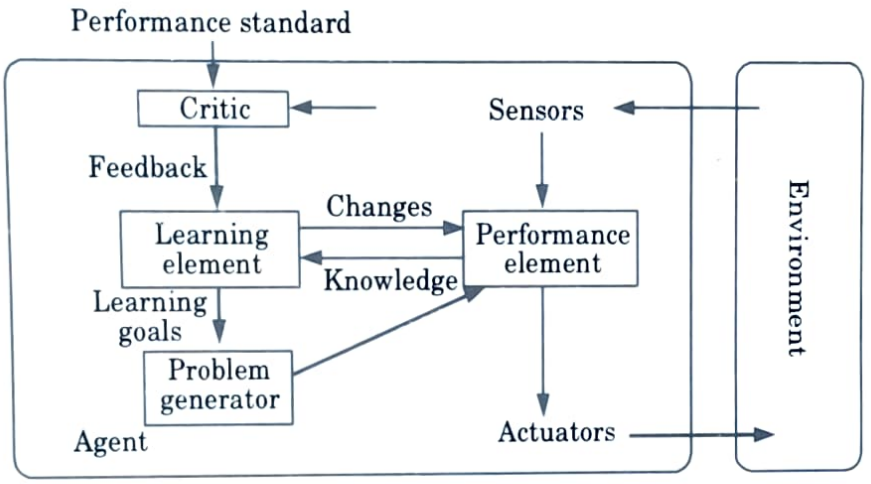


Fig. 1.10.1.

2. Four component of learning agent are :
   - **a. Learning element :**
     - i. The learning element responsible for making improvements.
     - ii. It uses feedback from the critic on how the agent is doing and determines how the performance elements should be modified in the future.
     - ii. The design of learning element depends on the design of performance element.
   - **b. Performance element :** It is responsible for selecting external action.
   - **c. Critic :**
     - i. It tells the learning elements how well the agent is doing with respect to a fixed performance standard.
     - ii. It is necessary because the percepts themselves provide no indication of the agent's success.
   - **d. Problem generator :**
     - i. Problem generator is responsible for suggesting actions that will lead to new and informative experiences.
     - ii. It also suggests exploratory actions.

Que 1.11. Explain in detail on the characteristics and applications of learning agents. **AKTU 2016-17, Marks 10**

Answer

Characteristics of learning agent :

1. **Situatedness :** When an agent receives some form of sensory input from its environment, it then performs some actions that change its environment in some way.
2. **Autonomy :** This agent characteristic means that an agent is able to act without direct intervention from humans or other agents. This type of agent has almost complete control over its own actions and internal state.
3. **Adaptivity :** This agent characteristic means that it is capable of reacting flexibly to changes within its environment. It is able to accept goal directed initiatives when appropriate and is also capable of learning from its own experiences, environment and interaction with others.
4. **Sociability :** This type of characteristic means that the agent is capable of interacting in a peer-to-peer manner with other agents or humans.

Applications of learning agent :

1. Clustering
2. Classification
3. Prediction
4. Search engines
5. Computer vision
6. Self-driving car
7. Recognition of gestures

PART-4

Problem Solving Approach to Typical AI Problems.

Questions-Answers

Long Answer Type and Medium Answer Type Questions

Que 1.12. What are the AI components that are required to solve problem ?

Answer

AI components that are required to solve problem are : There are six major components of an artificial intelligence system. They are solely responsible for generating desired results for particular problem. These components are as follows,

1. **Knowledge Representation :** It is used for representing necessary knowledge so as to generate knowledge base with the help of which AI system can perform tasks and generate results.
2. **Heuristic Searching Techniques :** While dealing with the problems the knowledge base keeps on growing and growing making it difficult to search in that knowledge base. To tackle this challenge, heuristic searching techniques can be used which can provide results efficiently in terms of time and memory usage.
3. **Artificial Intelligence Hardware :** Hardware must be efficient to accommodate and produce desire results. Hardware components include each and every machinery required spanning from memory to processor to communicating devices.
4. **Computer Vision and Pattern Recognition :** AI programs capture the inputs on their own by generating a real world scenario with the help of this component. Sufficient and compatible hardware enables better patterns gathering that makes a useful knowledgebase.
5. **Natural Language Processing :** This component processes or analysis written or spoken languages. Acquiring the word sequence and parsing sentence into computer is no just sufficient to gain knowledge about environment for AI systems. Natural Language processing plays vital role in understanding of domain of text to AI systems.
6. **Artificial Intelligence Language and Support Tools :** Artificial intelligence languages are almost similar to traditional software development programming languages with additional feature to capture human brain processes and logic as much as possible.

Que 1.13. Describe the formal description of the problem ?

Answer

Formal description of the problem is as follows :

1. **Explicit goal of the problem :** Goals help to organize behaviour of systems by limiting the objectives that the agent is trying to achieve. Goal formulation is based on the current situation and the agent's performance measure. It is first step towards problem solving.
2. **Implicit criteria for success :** That is how success is defined. That will be the ultimate thing system needs to achieve, which is the problem solution's output.
3. **Initial situation :** It means that what is going to be the start state of problem being solved.
4. **Ability to perform :** It tells how agents transform from one situation to another, how operations and rules are specified which change the states of the problem during solution process.

Que 1.14. Explain the two main kinds of problem formulation ?

Answer

Two main kinds of problem formulation :

1. **Incremental formulation :**
   i. It involves operators that augment the state description, starting with an empty state.
   ii. It generates many sequences.
   iii. Memory requirements is less as all states are not explored (exploration will be done till the goal is found).
2. **Complete state formulation :**
   i. In this initially we will have some basic configuration represented in initial state.
   ii. Here while doing any action first the conditions on the actions will be checked so that the configuration state after the action will be same legal state.
   iii. It takes up large memory as complete state space is generated. This formulation reduces number of sequences generated.

Que 1.15. What are the steps involved in problem solving agent ?

Answer

Problem solving agent achieves success by taking following approach to problem solution :

Step 1 : Goal setting

Agent set the goal by considering the environment.

Step 2 : Goal formulation

The goals set in step 1 are formalized in the frame work. The key activity in goal formulation is :

i. To observe current state.
ii. To tabulate agent's performance measures.

Step 3 : Problem formulation

i. After formulating goal, it is required to find out what will be the sequence of actions which generate goal state.
ii. Problem formulation is a way of looking at actions and states generated because of actions, which leads to success.

Step 4 : Search in unknown environment

i. If the task environment is unknown then agent first tries different sequence of actions and gathers knowledge (*i.e.*, learning). Then agent

gets known set of actions which leads to goal state. Thus agent search for describable sequence of actions this process is called as searching process.

ii. With knowledge of environment and goal state we can design a search algorithm. A search algorithm is a procedure which takes problem as input and returns its solution which is represented in the form of action sequence.

Step 5 : Execution phase

i. Once the solution is given by the search algorithm then the actions suggested by the algorithm are executed.

ii. This is the execution phase. Solution guides agent for doing the actions. After executing the actions, agent again formulate new goal.

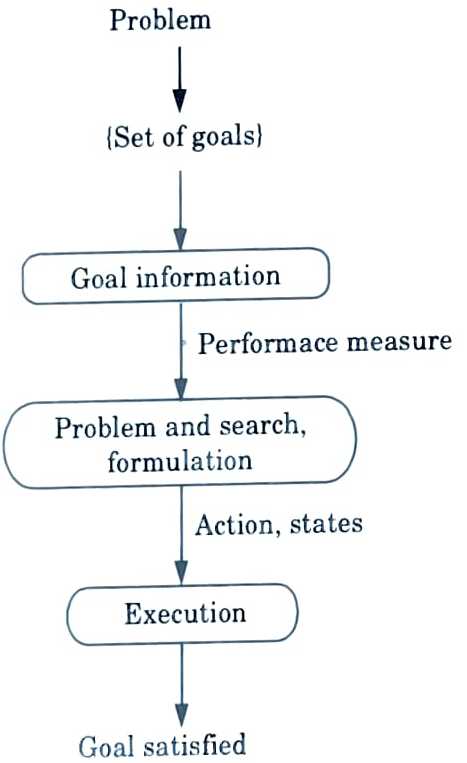


Fig. 1.15.1. Problem solving agent.

2
UNIT

Problem Solving Methods

CONTENTS

PART-1

Problem Solving Methods.

Questions-Answers

Long Answer Type and Medium Answer Type Questions

Que 2.1. What are the two categories of problem ?

Answer

We can distinguish problems in two categories :

1. **Toy problem :**
   - i. A toy problem is a problem which illustrates various problem solving methods.
   - ii. For the toy problem exact and precised description can be given.
   - iii. These problems provide basis for solving some real-life problems.
   - iv. They can be used by researchers to compare performance of algorithms.
   - v. For example : 8-queen puzzle, vacuum world, bali picker robot.
2. **Real world problem :**
   - i. A real-world problem is a problem which needs to be solved so that its solution can be utilized in practical life.
   - ii. They do not have well described, single specification.
   - iii. People do care about the solutions of real-world problem as they are benefited from it.
   - iv. For example : Route finding for a trip, travelling salesman problem, robot navigation, car reversing guide.

Que 2.2. Explain construction of state space.

Answer

Construction of state space :

1. The root of search tree is a search node corresponding to initial state. In this state only we can check if goal is reached.
2. If goal is not reached we need to consider another state. Such a process can be done by expanding from the current state by applying successor function which generates new state. From this we may get multiple states.

3. For each one of these, again we need to check goal test or else repeat expansion of each state.
4. The choice of which state to expand is determined by the search strategy.
5. It is possible that some state, surely, can never lead to goal state. Such a state we need not to expand. This decision is based on various conditions of the problem.

Que 2.3. What are various terminologies used in search tree ?

Answer

Terminologies used in search trees are :

1. **Node in a tree :** It is a book keeping data structure to represent the structure configuration of a state in a search tree.
2. **State :** It reflects world configuration. It is mapping of state and action to another new state.
3. **Fringe :** It is a collection of nodes that have been generated but not yet expanded.
4. **Leaf node :** Each node in fringe is leaf node (as it does not have further successor node).

Que 2.4. How performance of problem solving algorithm is evaluated ?

Answer

Problem solving algorithm's performance can be evaluated on the following four basic factors :

1. **Completeness :** Does the algorithm surely find a solution, if really the solution exists.
2. **Optimality :** Sometimes it happens that there are multiple solutions to a single problem. But the algorithm is expected to produce best solution among all feasible solution, which is called as optimal solution.
3. **Time complexity :** How much time the algorithm takes to find the solution.
4. **Space complexity :** How much memory is required to perform the search algorithm.

Que 2.5. Explain state space approach for solving any AI problem.

AKTU 2020-21, Marks 07

Answer

1. A state is a representation of problem elements at a given moment.
2. A State space is the set of all states reachable from the initial state.
3. A state space forms a graph in which the nodes are states and the arcs between nodes are actions.
4. In the state space, a path is a sequence of states connected by a sequence of actions.
5. The solution of a problem is part of the graph formed by the state space.
6. The state space representation forms the basis of most of the AI methods.
7. Its structure corresponds to the structure of problem solving in two important ways :
   a. It allows for a formal definition of a problem as per the need to convert some given situation into some desired situation using a set of permissible operations.
   b. It permits the problem to be solved with the help of known techniques and control strategies to move through the problem space until goal state is found.
8. To solve a particular problem, we need to build a system or a method which can generate required solution. Following four things are required for building such system.
   a. Define the problem precisely. This definition must precisely specify the initial situation (input).
   b. Analyse the problem. To identify those important features which can have an immense impact on the appropriateness.
   c. Isolate and represent the task knowledge that is necessary to solve the problem.
   d. Choose the best problem solving technique and apply it to the particular problem.

PART-2

Search Strategies, Uninformed, Informed, Heuristic.

Questions-Answers

Long Answer Type and Medium Answer Type Questions

Que 2.6. Define searching process.

OR

What are the different parameters used to evaluate a search technique ? **AKTU 2014-15, Marks 05**

OR

List the criteria to measure the performance of different search strategies. **AKTU 2018-19, Marks 05**

Answer

Searching :

1. Searching is the sequence of steps that transforms the initial state to the goal state.
2. The process of search includes :
   a. Initial state description of the problem.
   b. A set of legal operators that change the state.
   c. The final or goal state.

Following are the parameters used to evaluate a search technique :

1. **Completeness :** By completeness, the algorithm finds an answer in some finite time. So, the algorithm is said to be complete if it is guaranteed to find a solution, if there is one.
2. **Space and time complexity :** With space and time complexity, we address the memory required and the time factor in terms of operations carried out.
3. **Optimality :** Optimality tells us how good the solution is. So, an algorithm or a search process is said to be optimal, if it gives the best solution.

Que 2.7. Describe the types of search strategies.

Answer

Types of search strategies :

Uninformed search :

1. Uninformed search means that they have no additional information about states beyond that provided in the problem definition.
2. An uninformed search should proceed in a systematic way by exploring nodes in some predetermined order or simply by selecting nodes at random.
3. Uninformed search is also called Brute Force Search or Blind Search or Exhaustive Search.
4. It is of following types :
   i. Breadth First Search
   ii. Depth First Search
   iii. Uniform Cost search

Informed search :

1. Informed search algorithm contains an array of knowledge such as how far we are from the goal, path cost, how to reach to goal node, etc. This knowledge helps agents to explore less to the search space and find more efficiently the goal node.
2. Informed search methods often depend on the use of heuristic information.
3. Informed search is also known as heuristic search.
4. Types of informed search are :
   i. Hill climbing
   ii. Best First Search
   iii. A* algorithm

Que 2.8. Differentiate between informed search and uninformed search.

Answer

S. No.	Informed search	Uninformed search
1.	It uses knowledge to find the steps to the solution.	No use of knowledge.
2.	It is highly efficient as consumes less time and cost.	Efficiency is mediatory.
3.	Cost is low.	Cost is comparatively high.
4.	It finds solution more quickly.	Its speed is slow.
5.	For example : Best first search, A* algorithm	For example : Breadth First Search, Depth First Search

Que 2.9. Write a short note on

i. Breadth First Search (BFS)
ii. Depth First Search (DFS)

OR

Explain DFS algorithm with suitable example.

AKTU 2020-21, Marks 07

Answer

i. Breadth First Search (BFS) :

1. Breadth First Search (BFS) is an algorithm for traversing or searching tree.

2. It starts at the tree root (or some arbitrary node of a graph, sometimes referred to as a 'search key'), and explores all of the neighbour nodes at the present depth prior to moving on to the nodes at the next depth level.

Algorithm :

1. Create a variable called NODE-LIST and set it to the initial state.
2. Until a goal state is found or NODE-LIST is empty :
   a. Remove the first element from NODE-LIST and call it *E*. If NODE-LIST was empty, quit.
   b. For each way that each rule can match the state described in *E* do :
      i. Apply the rule to generate a new state.
      ii. If the new state is a goal state, quit and return this state.
      iii. Otherwise, add the new state to the end of NODE-LIST.

ii. Depth First Search (DFS) :

1. Depth First Search (DFS) is an algorithm for traversing or searching tree.
2. The algorithm starts at the root node (selecting some arbitrary node as the root node in the case of a graph) and explores as far as possible along each branch before backtracking.

Algorithm :

1. If the initial state is a goal state, quit and return success.
2. Otherwise, do the following until success or failure is signaled :
   a. Generate a successor, *E*, of the initial state. If there are no more successors, signal failure.
   b. Call Depth First Search with *E* as the initial state.
   c. If success is returned, signal success. Otherwise continue in this loop.

Que 2.10. Write the advantages and disadvantages of BFS and DFS.

Answer

Breadth First Search (BFS) :

Advantages :

1. If there is more than one solution for a given problem, then BFS provides the minimal solution which requires the least number of steps.

Disadvantages :

1. It requires lots of memory since each level of the tree must be saved into memory to expand the next level.
2. BFS needs lots of time if the solution is far away from the root node.

Depth First Search (DFS) :

Advantages :

1. DFS requires very less memory as it only needs to store a stack of the nodes on the path from root node to the current node.
2. It takes less time to reach to the goal node.

Disadvantages :

1. There is the possibility that many states keep re-occurring, and there is no guarantee of finding the solution.
2. DFS algorithm goes for deep down searching and sometime it may go to the infinite loop.

Que 2.11. Differentiate between Breadth First Search (BFS) and Depth First Search (DFS).

Answer

S. No.	Breadth First Search (BFS)	Depth First Search (DFS)
1.	BFS uses queue data structure for finding the shortest path.	DFS uses stack data structure for finding the shortest path.
2.	BFS is more suitable for searching vertices which are closer to the given source.	DFS is more suitable when there are solutions away from source.
3.	BFS consider all neighbours first and therefore not suitable for decision making trees used in games or puzzles.	DFS is more suitable for game or puzzle problems.
4.	The time complexity of BFS is $O(V + E)$, where V stands for vertices and E stands for edges.	The time complexity of DFS is also $O(V + E)$, where V stands for vertices and E stands for edges.

Que 2.12. Write a short note on generate and test algorithm.

Answer

Generate-and-test is a search algorithm which uses depth first search technique. It assures to find solution in systematic way.

Algorithm for Generate-and-Test :

1. Generate a possible solution which can either be a point in the problem space or a path from the initial state.

2. Test to see if this possible solution is a real (actual) solution by comparing the state reached with the set of goal states.
3. If it is real solution then return the solution otherwise repeat from state 1.

Que 2.13. Write down the algorithm for best first search.

Answer

Algorithm for best first search :

1. Use two ordered lists OPEN and CLOSED.
2. Start with the initial node 'n_1' and put it on the ordered list OPEN.
3. Create a list CLOSED. This is initially an empty list.
4. If OPEN is empty then exit with failure.
5. Select first node on OPEN. Remove it from OPEN and put it on CLOSED. Call this node n.
6. If 'n' is the goal node exit. The solution is obtained by tracing a path backward along the arcs in the tree from 'n' to 'n_1'.
7. Expand node 'n_1'. This will generate successors. Let the set of successors generated, be S. Create arcs from 'n' to each member of S.
8. Reorder the list OPEN, according to the heuristic and go back to step 4.

Que 2.14. Prove that breadth first search and depth first search are the special cases of best first search.

AKTU 2017-18, Marks 10

Answer

1. Best first search is a combination of depth first and breadth first searches.
2. Depth first is good because a solution can be found without computing all nodes and breadth first is good because it does not get trapped in dead ends.
3. The best first search allows us to switch between paths thus gaining the benefit of both approaches. At each step the most promising node is chosen.
4. If one of the nodes chosen generates nodes that are less promising it is possible to choose another at the same level and in effect the search changes from depth to breadth.
5. If on analysis these are no better than the search method reverts to the descendants of the first choice and proceeds the backtracking as it were.

Que 2.15. **What is heuristic function ? Differentiate between blind search and heuristic search strategies.** **AKTU 2018-19, Marks 05** **AKTU 2019-20, Marks 07**

Answer

Heuristic function :

1. A heuristic function is a function that ranks alternatives in search algorithms at each branching step based on available information to decide which branch to follow.
2. It takes the current state of the agent as its input and produces the estimation of how close agent is from the goal.
3. The heuristic method might not always give the best solution, but it guarantees to find a good solution in reasonable time.
4. Heuristic function estimates how close a state is to the goal.
5. It calculates the cost of an optimal path between the pair of states. The value of the heuristic function is always positive.

Difference :

S. No.	Blind search strategies	Heuristic search strategies
1.	Blind search technique has access only to the problem definition.	Heuristic search technique has access to heuristic function as well as problem definition.
2.	Blind search is less efficient.	Heuristic search is more efficient.
3.	Every action is equally good in blind search.	Every action is not equally good in heuristic search.
4.	Many problems are not solved by blind search.	Most of the problems are solved by heuristic search.
5.	Blind search is known as uninformed search.	Heuristic search is known as informed search.
6.	Blind search use more computation.	Heuristic search use less computation.
7.	Blind search techniques include Breadth-first, Depth-first search, etc.	Heuristic search techniques include Best-first, A*, etc.

Que 2.16. **Describe AO* search technique.**

AKTU 2015-16, Marks 05

Answer

AO* algorithm :

1. Initialise the graph to start node.
2. Traverse the graph following the current path accumulating nodes that have not yet been expanded or solved.
3. Pick any of these nodes and expand it and if it has no successors call this value FUTILITY otherwise calculate only f' for each of the successors.
4. If f' is 0 then mark the node as SOLVED.
5. Change the value of f' for the newly created node to reflect its successors by back propagation.
6. Wherever possible use the most promising routes and if a node is marked as SOLVED then mark the parent node as SOLVED.
7. If starting node is SOLVED or value greater than FUTILITY, stop, else repeat from 2.

PART-3

Local Search Algorithm and Optimization Problems, Searching with Partial Observation.

Questions-Answers

Long Answer Type and Medium Answer Type Questions

Que 2.17. **Explain about the hill climbing algorithm with its drawback and how it can be overcome ?**

AKTU 2015-16, Marks 10

OR

Discuss the problems of hill climbing algorithm.

AKTU 2018-19, Marks 10

OR

Explain about the Hill climbing algorithm with its drawback and how it can be overcome ?

AKTU 2019-20, Marks 07

Answer

1. The hill climbing search algorithm is simply a loop that continually moves in the direction of increasing values, that is, uphill (the goal).

2. It terminates when it reaches a "peak" where no neighbour has a higher value.
3. The algorithm does not maintain a search tree, so the current node data structure only records the state and its objective function value.

Hill climbing algorithm :

1. Evaluate the initial state. If it is also a goal state, then return it and quit. Otherwise, continue with the initial state as the current state.
2. Loop until a solution is found or until there are no new operators left to be applied in the current state.
   a. Select an operator that has not yet been applied to the current state and apply it to produce a new state.
   b. Evaluate the new state.
      i. If it is goal state, then return it and quit.
      ii. If it is not a goal state but it is better than the current state, then make it the current state.
      iii. If it is not better than the current state, then continue in the loop.

Drawbacks of hill climbing algorithm :

1. **Local maxima :**
   i. A local maximum is a peak that is higher than each of its neighbouring states, but lower that the global maximum.
   ii. Hill climbing algorithms that reach the vicinity of a local maximum will be drawn upwards towards the peak, but will then be stuck with nowhere else to go.
2. **Plateau :**
   i. A plateau is a flat area of the search space in which a whole set of neighbouring states has the same value.
   ii. A hill climbing search might be unable to find its way off the plateau.
3. **Ridges :** Ridges result in a sequence of local maxima that is very difficult for greedy algorithms to navigate.

To overcome the drawbacks of hill climbing algorithm :

1. **Local maxima :**
   i. Backtrack to some earlier node and try going in a different direction.
   ii. To implement this strategy, maintain a list of paths almost taken and go back to one of them if the path that was taken leads to a dead end.
2. **Plateau :**
   i. Make a big jump in some direction to try to get to a new section of the search space.

ii. If the only rules available describe single small steps, apply them several times in the same direction.

3. **Ridges :**

i. Apply two or more rules before doing the test.

ii. This corresponds to moving in several directions at once.

Que 2.18. Discuss the problem of water jug with heuristic search techniques. **AKTU 2015-16, Marks 10**

Answer

Water jug problem :

1. You are given two jugs, a 4 litres one and a 3 litres one. Neither has any measuring marker on it. There is a pump that can be use to fill the jugs with water. How can you get exactly 2 litres of water into the 4 litres jug ?
2. The state space for this problem can be represented by ordered pairs of integers (x, y) such that $x = 0$, represents the quantity of water in the 3 litres jug.
3. The start state is (0, 0)
4. The goal state is $(2, n)$ for any value of n.

Production rules :

1.	$(x, y) \rightarrow (4, y)$ If $x < 4$	Fill the 4 litres jug.
2.	$(x, y) \rightarrow (x, 3)$ If $y < 3$	Fill the 3 litres jug.
3.	$(x, y) \rightarrow (x - d, y)$ If $x > 0$	Pour some water out of the 4 litres jug.
4.	$(x, y) \rightarrow (x, y - d)$ If $y > 0$	Pour some water out of the 3 litres jug.
5.	$(x, y) \rightarrow (0, y)$ If $x > 0$	Empty the 4 litres jug on the ground.
6.	$(x, y) \rightarrow (x, 0)$ If $y > 0$	Empty the 3 litres jug on the ground.
7.	$(x, y) \rightarrow (4, y - (4 - x))$ If $x - y \geq 0$ and $y > 0$	Pour water from the 3 litres jug into the 4 litres jug until the 4 litres jug is full.
8.	$(x, y) \rightarrow (x - (3 - y), 3)$ If $x + y \geq 3$ and $x > 0$	Pour water from the 4 litres jug into the 3 litres jug until the 3 litres jug is full.

9.	(x, y) →(x – y, 0) If x + y≤ 4 and y > 0	Pour all the water into the 4 litres jug.
10.	(x, y) → (0, x + y) If x + y ≤ 3 and x > 0	Pour all the water from the 4 litres jug into the 3 litres jug.
11.	(0, 2) → (2, 0)	Pour the 2 litres from the 3 litres jug into the 4 litres jug.
12.	(2, y) → (0, y)	Empty the 2 litres in the 4 litres jug on the ground.

PART-4

Constraint Satisfaction Problems, Constraint Propagation.

Questions-Answers

Long Answer Type and Medium Answer Type Questions

Que 2.19. Explain the constraint satisfaction procedure to solve the cryptarithmetic problem.

CROSS + ROADS = DANGER. **AKTU 2016-17, Marks 10**

Answer

Constraint satisfaction is a search procedure that operates in a space of constraint sets. The initial state contains the constraints that are originally given in the problem description.

The general form of the constraint satisfaction procedure is :

1. Select an unexpected node of the search graph.
2. To generate all possible new constraints, apply the constraint inference rule to the selected node.
3. If the set of constraints contain a contradiction, then report that this path is a dead end.
4. To report success, if the set of constraints, a contradiction describes a complete solution.
5. If neither a contradiction nor a complete solution has been found then apply the problem space rules to generate new partial solutions that are consistent with the current set of constraints. Insert these partial solutions into the search graph.

Numerical :

Constraints :

1. D = 1
2. 2S = R
3. C + R > 9 or 8
4. S + 1 = E = > S = E – 1

   Hence, after guessing

CROSS	96233
+ ROADS	+ 62513
.....	.....
DANGER	158746

Que 2.20. Discuss constraint satisfaction problem with an algorithm for solving a cryptarithmetic problem

BASE
+ BASE
GAMES

AKTU 2020-21, Marks 07

Answer

1. Initial guess $G = 1$ because the sum of two digit can generate at most a carry 1.
2. If $G = 1$ then B should be either 8 or 9. $G + B$ gives a two digit number this also considers the carry digit.
3. Let us consider two equation

   If no carry $E + L = S$ ...(2.19.1)

   If carry $E + L = S + 10$ ...(2.19.2)

   $E = S - L + 10$
4. Now, $S + L = E$ ...(2.19.3)

   Put value of E in eq. (2.19.3)

   $S + L = S - L + 10$

   $2L = 10$

   $L = 5$
5. From eq. (2.19.1)

   $S - E = L$

   $S - E = 5$
6. Now, we can try all the combination of S and E whose difference will be S.

(5, 0), (7, 2), (8, 3), (6, 1), (9, 4)

S, cannot be 5 because $L = 5$, E cannot be 1 because $G = 1$,

6. Now $B + B = A$ + carry, possible value of B will be 5, 6, 7, 8, 9, B cannot be 5, because 5 is assigned to L.

Let $(S, E) = (7, 2)$ and $B = 6$.

$B + B = 12 \Rightarrow A = 2$ which contradicts with E discard (7, 2).

Let $(S, E) = (8, 3)$ and $B = 7$.

$B + B = 14 \Rightarrow A = 4, B = 7, S = 8, E = 3, G = 1$

BASE	7 4 8 3
+ BASE	+ 7 4 5 5
GAMES	1 4 9 3 8

Que 2.21. Solve the following CSP problem of crypt arithmetic.

Problem :

SEND
+ MORE
.....
MONEY

AKTU 2017-18, Marks 05

AKTU 2015-16, Marks 10

Answer

1. Initial guess, $M = 1$ because the sum of two single digits can generate at most a carry of '1'.
2. If $M = 1$, then S should be either 8 or 9 because $S + M$ gives a two-digit number. This also considers the carry digit.
3. When $M = 1$ and $S = 8/9$, the two digit result of $M + S$ can either be 10 or 11. That is, O will be either 0 or 1. But, 1 is already assigned to M so it can not be assigned to any other digit. Thus, $O = 0$, $(M + S) = 10$. S can be 8/9 depending on the carry.
4. Now, $E + 0 = N$, which is only possible if there's a carry of 1 because otherwise, $E + 0 = E$. Thus, $N = E + 1$ and $C2 = 1$.
5. So far, we have $M = 1$, $S = 8/9$, $O = 0$, $C2 = 1$. We are struck here because we do not know the value of E. Thus, we will try different possible values of E.
6. For $E = 5$, $N = 6$, $(N + R) = 15$, $R = 8/9$, $(D + 2) = Y$. Again, we do not know the value of $C1$. So, we will assume it.
7. For $C1 = 1$, $R = 8$, $S = 9$, $(D + 5) = 10 + Y$. Maximum value of D can be 7. If $D = 7$, $Y = 2$.
8. So, $M = 1$, $S = 9$, $O = 0$, $E = 5$, $N = 6$, $R = 8$, $D = 7$, $E = 5$, $Y = 2$.

Thus, we have our probable solution as follows :

SEND	9567
+ MORE	+ 1085
MONEY	10652

Que 2.22. Explain constraint propagation. What is constraint propagation with arc consistency ?

Answer

Constraint propagation :

1. A combined approach of heuristic plus forward checking gives more reliable, accurate and efficient results than a singular approach.
2. The forward checking propagates information from designed to unassigned variable but cannot avoid or detect all failure.
3. Constraint propagation repeatedly enforces constraints locally.
4. The sides of arc consistency provide a fast method of constraint propagation that is substantially stronger than forward checking. Here "arc" refers to a directed arc in the constraint graph.

Constraint propagation using arc consistency :

1. It is fast method of constraint propagation.
2. $X \rightarrow Y$ is consistent if (for every value of X there is source allowed value Y. For example $[V_2 \rightarrow V_1$, is consistent iff V_1 = Red, V_2 = Blue] (*i.e.*, for every value of x in X there is some allowed value y in Y). This is directed property example – $[V_1 = V_2]$

   $V_1 = V_2$ is consistent iff

   V_1 = Red and V_2 = Blue
3. As directed arcs between variables represent the domains of specified variables, they are consistent with each other.
4. Constraint propagation can be applied as preprocessing or propagation step :
   i. Before search-preprocessing.
   ii. After search-propagation.
5. The procedure for maintaining arc consistency can be applied repeatedly.

PART-5

Backtracking Search.

Questions-Answers

Long Answer Type and Medium Answer Type Questions

Que 2.23. Write a short note on backtracking.

AKTU 2015-16, Marks 05

Answer

1. The backtracking is an algorithmic method to solve a problem with an additional way.
2. It uses a recursive approach to explain the problems. The backtracking is needed to find all possible combination to solve an optimization problem.
3. It is a systematic way of trying out different sequences of decisions until we find one that works.
4. It is considered an important technique to solve constraint satisfaction issues and puzzles. It is also considered a great technique for parsing and also forms the basis of many logic programming languages.

Que 2.24. Write down backtracking search algorithm.

Answer

1. Consider a CSP problem.
2. Apply backtracking search. If backtracking search successful returns a solution else, a failure state which return a procedure recursive-backtracking.
3. Procedure recursive-backtracking starts with empty set and takes input as CSP problem.

   If complete assignment possible or assignment done then return actual assignment.
4. Variable is assigned a specific value.
5. The relative constraint is a set which is taken as input.
6. If value is complete and consistent according to constraints then assign value to variable, add that to list.
7. Call the recursive backtracking until result or failure is reached.
8. Every time recursive-backtracking sustains result (*i.e.*, assignment).
9. If result is failure then remove variable with specific value from assignment. It will return failure status.

PART-6

Game Playing, Optimal Decisions in Games.

Questions-Answers

Long Answer Type and Medium Answer Type Questions

Que 2.25. Why is game playing good candidate of AI ? Explain.

AKTU 2014-15, Marks 05

Answer

1. A good candidate for AI has following characteristics :
   i. It contains a large amount of domain-specific knowledge.
   ii. It contains computationally complex problems.
   iii. It can be developed as a repository for the knowledge of several experts.
2. Let us consider an example of game playing, an intelligent system that plays chess.
   i. The rules of chess are easy to learn, but to play this game at an expert level is not easy because it has 10^{120} possible games. This 10^{120} possible games of chess satisfy the first characteristics of good candidate for AI *i.e.*, large amount of domain-specific knowledge.
   ii. These 10^{120} possible games of chess have equally large and complex moves by various chess pieces (*i.e.*, pawns, rooks, king etc). These are computationally complex problems which cannot be solved by straightforward algorithms. This satisfies the second characteristics of good candidate for AI.
   iii. The chess program is build based on the inputs from several expert chess players. It has enormous amount of knowledge about chess (domain-specific knowledge) that it uses as part of its decision making process. This satisfies the third characteristics of good candidate for AI.

Hence we can say that game playing is good candidate for AI.

Que 2.26. Give an example of game tree. What is the purpose of minimax procedure in a game tree ?

AKTU 2014-15, Marks 05

Answer

1. A game tree is a directed graph whose nodes are positions in a game and whose edges are moves.
2. The complete game tree for a game is the game tree starting at the initial position and containing all possible moves from each position.
3. Fig. 2.26.1 shows the first two levels in the game tree for tic-tac-toe.

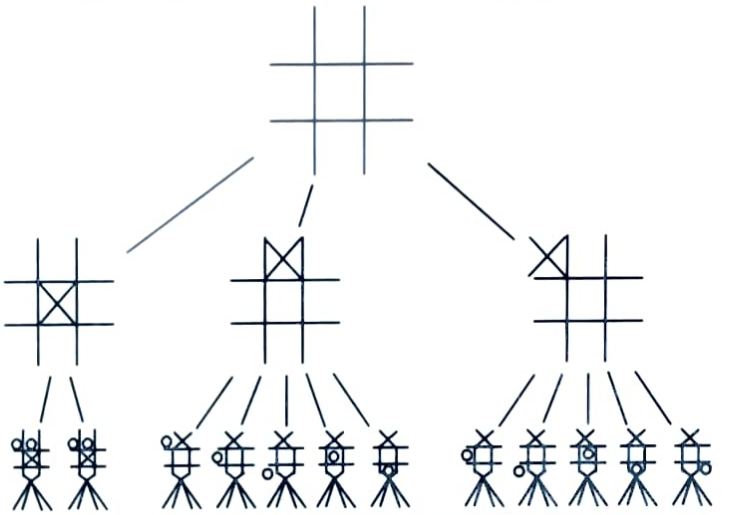


Fig. 2.26.1. The first two levels of the game tree for tic-tac-toe.

4. The rotations and reflections of positions are equivalent, so the first player has three choices of move : in the center, at the edge, or in the corner.
5. The second player has two choices for the reply if the first player played in the center, otherwise five choices. And so on.
6. The number of leaf nodes in the complete game tree is the number of possible different ways the game can be played.
7. For example, the game tree for tic-tac-toe has 255,168 leaf nodes.

Purpose of minimax procedure :

1. For many complex games such as chess, search to termination is impossible, *i.e.*, a win or draw cannot be obtained.
2. Our goal in searching such a game tree might be to find a good first move.
3. This good first move can be extracted by minimax procedure. This is the purpose of minimax procedure in a game tree.

Que 2.27. Explain various strategies of game playing.

Answer

Game playing strategies :

1. A pure strategy provides a complete definition of how a player will play a game. In particular, it determines the move a player will make for any

situation they could face. A player's strategy set, is the set of pure strategies available to that player.

2. A mixed strategy is an assignment of a probability to each pure strategy. This allows for a player to randomly select a pure strategy. Since probabilities are continuous, there are infinitely many mixed strategies available to a player, even if their strategy set is finite.

Que 2.28. Explain Min-Max algorithm with example.

AKTU 2016-17, Marks 05

Answer

Min-Max algorithm :

Step 1: Set FINAL_VALUE to be minimum as possible.

Step 2 : If limit of search has been reached, then FINAL_VALUE = GOOD_VALUE of the current position.

Step 3 : Else do.

Step 3.1 : Generate the successors of the position.

Step 3.2 : Recursively call MIN-MAX again with the present position with depth incremented by unity.

Step 4 : Evaluate the GOOD_VALUE.

Step 5 : If GOOD_VALUE > FINAL_VALUE then FINAL_VALUE = GOOD_VALUE.

For example :

1. Consider a game which has four final states and paths to reach final state are from root to four leaves of a perfect binary tree as shown Fig. 2.28.1.

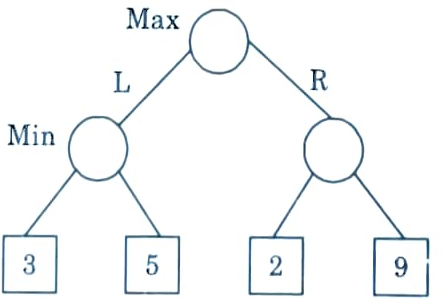


Fig. 2.28.1.

2. Assume that maximizing player get the first chance to move, *i.e.*, maximizer at the root and opponent at next level.

3. As this is a backtracking based algorithm, it tries all possible moves, then backtracks and makes a decision :

i **Maximizer goes LEFT :** It is now the minimizers turn. The minimizer now has a choice between 3 and 5. Being the minimizer it will definitely choose the least among both, that is 3.

ii **Maximizer goes RIGHT :** It is now the minimizers turn. The minimizer now has a choice between 2 and 9. He will choose 2 as it is the least among the two values.

4. Being the maximizer he would choose the larger value that is 3. Hence the optimal move for the maximizer is to go LEFT and the optimal value is 3.
5. Fig. 2.28.2 shows two possible scores when maximizer makes left and right moves.

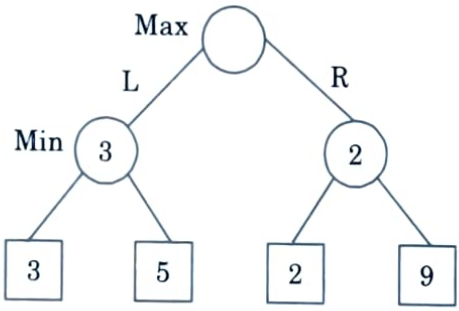


Fig. 2.28.2.

Que 2.29. What is adversarial search ? Write the steps for game problem formulation. State and explain minimax algorithm with tic-tac-toe game. **AKTU 2019-20, Marks 07**

Answer

Adversarial search :

1. Adversarial search is a search, where we examine the problem which arises when we try to plan ahead of the world and other agents are planning against us.
2. The environment with more than one agent is termed as multi-agent environment, in which each agent is an opponent of other agent and playing against each other. Each agent needs to consider the action of other agent and effect of that action on their performance.
3. So, searches in which two or more players with conflicting goals are trying to explore the same search space for the solution are called adversarial searches, often known as Games.

Steps for game problem formulation : Problem formulation involves deciding what actions and states to consider for the given goal. A problem can be defined formally by five components :

1. The initial state of the agent.
2. The possible actions available to the agent, corresponding to each of the state the agent resides in.
3. The transition model describing what each action does.
4. The goal test, determining whether the current state is a goal state.
5. The path cost function, which determines the cost of each path, which is reflecting in the performance measure.

Minimax algorithm : Refer Q. 2.26, Page 2–19A, Unit-2.

PART-7

Alpha Beta Pruning, Stochastic Games.

Questions-Answers

Long Answer Type and Medium Answer Type Questions

Que 2.30. How branch and bound techniques could be used to find the shortest path solution to the travelling salesman problem. Discuss. **AKTU 2017-18, Marks 05**

Answer

The branch and bound strategy is a system for solving a sequence of subproblems each of which may have multiple possible solutions and where the solution chosen for one sub-problem may affect the possible solutions of later sub-problems.

Principle :

1. Suppose it is required to minimize an objective function.
2. Suppose that we have a method for getting a lower bound on the cost of any solution among those in the set of solutions represented by some subset.
3. If the best solution found so far costs less than the lower bound for this subset :

   Let S be some subset for solution. Let

   $L(S)$ = a lower bound on the cost of any solution belonging to S

   C = cost of the best solution found so far

   If $C \leq L(S)$, there is no need to explore S because it does not contain any better solution.

If $C > L(S)$, then we need to explore S because it may contain a better solution.

A lower bound for a TSP :

Cost of any tour = $\frac{1}{2}\sum_{v \in V}$ (Sum of the costs of the two tour edges adjacent to v)

Now,

The sum of the two tour edges adjacent to a given vertex $v \geq$ sum of the two edges of least cost adjacent to v.

Therefore,

Cost of any tour $\geq \frac{1}{2}\sum_{v \in V}$ (Sum of the costs of the two least cost edges adjacent to v).

Example of complete graph with five vertices :

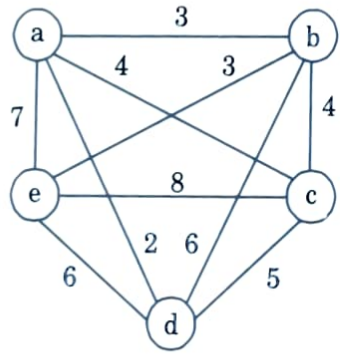


Fig. 2.30.1.

Node	Least Cost Edges	Total Cost
a	(a, d), (a, b)	5
b	(a, b), (b, e)	6
c	(c, b), (c, a)	8
d	(d, a), (d, c)	7
e	(e, b), (e, d)	9

Thus, a lower bound on the cost of any tour

$$= \frac{1}{2}(5 + 6 + 8 + 7 + 9) = 17.5$$

Que 2.31. **Explain Min-Max procedure. Describe alpha-beta pruning and give the other modifications to the Min-Max procedure to improve its performance.** **AKTU 2015-16, Marks 15**

Answer

Min-Max procedure :

1. The min-max algorithm removes unwanted nodes and selects the optimal node for progressing further.
2. The min-max algorithm computes the min-max decision from the current state.
3. It uses a simple recursive computation of the min-max values of each successor state.

Min-Max algorithm : Refer Q. 2.28, Page 2–21A, Unit-2.

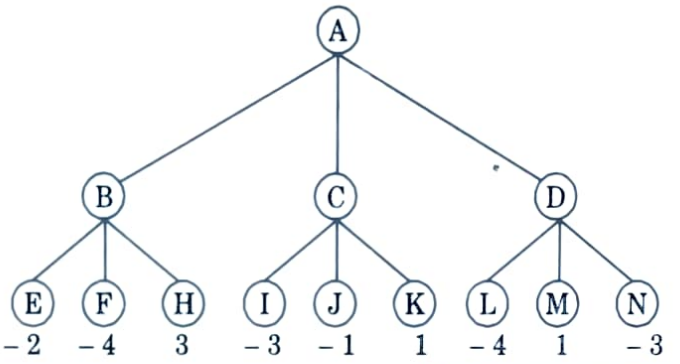
Alpha-beta pruning :

1. Alpha-beta pruning is a modified version of the min-max algorithm. It is an optimization technique for the min-max algorithm.
2. In min-max search algorithm, the number of nodes (game states) that it has to examine is exponential in depth of the tree. Now we cannot eliminate the exponent completely, but we can cut it to half.
3. There is a technique by which without checking each node of the game tree we can compute the correct min-max decision, and this technique is called pruning. This involves two threshold parameter alpha and beta for future expansion, so it is called alpha-beta pruning.
4. Alpha-beta pruning can be applied at any depth of a tree, and sometime it not only prunes the tree leaves but also entire sub-tree.
5. The two parameters can be defined as :
   a. **Alpha :** The best (highest-value) choice we have found so far at any point along the path of Maximizer. The initial value of alpha is $-\infty$.
   b. **Beta :** The best (lowest-value) choice we have found so far at any point along the path of Minimizer. The initial value of beta is $+\infty$.

Modifications to min-max : There are some heuristic search methods other than alpha-beta pruning method which are used to improve the performance of min-max procedure. They are :

1. Greedy hill climbing method
2. Artificial immune algorithm

Que 2.32. **Explain alpha-beta pruning ? Solve the following question.**

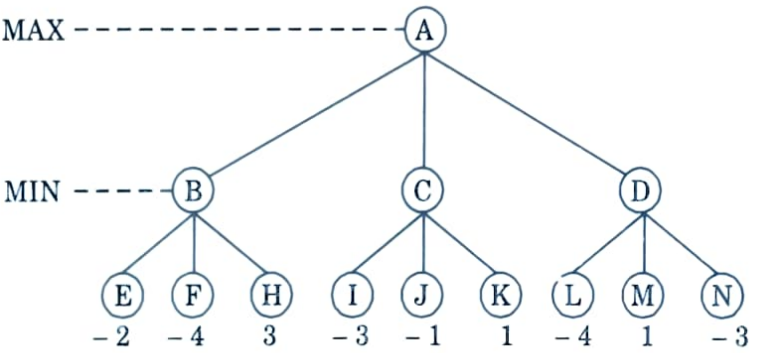


AKTU 2020-21, Marks 07

Answer

a. Alpha beta pruning : Refer Q. 2.31 , Page 2–25A , Unit-2.

b. Numerical :



Let, $\alpha = \max = -\infty$

$\beta = \min = \infty$

At A, A is maximizer, $\alpha = -\infty$, $\beta = \infty$. We use Depth First Search and go to B.

At B, B is minimizer, initially $\alpha = -\infty$, $\beta = \infty$. Value of β will change. $\beta = \min(-2, -4, 3) = -4$.

At B, $\alpha = -\infty$, $\beta = -4$

Now at A, value of α will change

$$\alpha = \max(-\infty, -\infty, -4) = -4, \beta = +\infty$$

Initially at C, $\alpha = -4$, $\beta = \infty$

Now, C is minimizer, so value of β will change

$$\beta = \min(-3, -1, 1) = -3, \alpha = -4$$

Now again at A value of α will change

$$\alpha = \max(-3, -4, -4) = -3, \beta = \infty$$

Initially at D, $\alpha = -4$, $\beta = \infty$

Now, D is minimizer, so value of β will change

$$\beta = \min(-4, 1, -3) = -4, \alpha = -4$$

Que 2.33. Write a short note on stochastic game.

Answer

1. A stochastic game is a collection of normal-form games that the agents play repeatedly.
2. The particular game played at any time depends probabilistically on the previous game played and the actions of the agents in that game.
3. Like a probabilistic finite state machines in which the states are the games and the transition labels are joint action-payoff pairs.
4. A stochastic game also called as Markov game is defined by :
   i. A finite set Q of states (games).
   ii. A set of strategies $Si(x)$ for each player for each state $x \in X$.
   iii. A set $N = (1, ..., n)$ of agents.
   iv. For each agent i, a finite set AI of possible actions.
   v. A transition probability function $P : Q \times A_1 \times ... \times A_n \times Q \rightarrow [0, 1]$
   $P(q, a_1, ..., a_n, q)$ = Probability of transitioning to state q
   if the action profile $(a_1, ..., a_n)$ is used in state q
   vi. A set of rewards dependant on the state and the actions of the other players : $u_1(x, S_1, S_2)$.
   vii. For each agent i, a real-valued payoff function.
   viii. $ri : Q \times A_1 \times, ..., A_n$ -> R (set of real numbers).
   ix. Each stage game is played of a set of discrete times t.

Que 2.34. What are the strategies for solving stochastic games ?

Answer

Strategies for solving stochastic games :

1. For agent i, a deterministic strategy specifies a choice of action for i at every stage of every possible history.
2. A mixed strategy is a probability distribution over deterministic strategies.
3. Several restricted classes of strategies :
   i. As in extensive-form games, a behavioral strategy is a mixed strategy in which the mixing take place at each history independently.
   ii. A Markov strategy is a behavioral strategy such that for each time t, the distribution over actions depends only on the current state. But the distribution may be different at time t than at time t.
   iii. A stationary strategy is a Markov strategy in which the distribution over actions depends only on the current state (not on the time t).

UNIT 3

Knowledge Representation

CONTENTS

PART-1

First Order Predicate Logic.

Questions-Answers

Long Answer Type and Medium Answer Type Questions

Que 3.1. Describe first order logic in artificial intelligence.

Answer

1. First order logic is also known as Predicate logic or First order predicate logic. First order logic is a powerful language that develops information about the objects in a more easy way and can also express the relationship between those objects.
2. First order logic is an extension to propositional logic.
3. First order logic is sufficiently expressive to represent the natural language statements in a concise way.
4. First-order logic (like natural language) does not only assume that the world contains facts like propositional logic but also assumes objects, relations, function.
5. As a natural language, first-order logic also has two main parts syntax and semantics.

Que 3.2. Explain inference rules in first order logic.

OR

Describe the rules of inference in first order predicate logic with suitable example. **AKTU 2019-20, Marks 07**

Answer

Inference rules for first order logic :

1. **Universal generalization :**
   i. Universal generalization is a valid inference rule which states that if premise $P(c)$ is true for any arbitrary element c in the universe of discourse, then we can have a conclusion as $\forall \times P(x)$.
   ii. It can be represented as :

$$= \frac{P(c)}{\forall \times P(x)}$$

iii. This rule can be used if we want to show that every element has a similar property.

2. Universal instantiation :

i. Universal instantiation is also called as universal elimination or UI is a valid inference rule. It can be applied multiple times to add new sentences.

ii. As per UI, we can infer any sentence obtained by substituting a ground term for the variable.

iii. The UI rule state that we can infer any sentence $P(c)$ by substituting a ground term c (a constant within domain x) from $\forall \times P(x)$ for any object in the universe of discourse.

iv. It can be represented as :

$$= \frac{\forall \times P(x)}{P(c)}$$

3. Existential instantiation :

i. Existential instantiation is also called as Existential elimination, which is a valid inference rule in first order logic.

ii. It can be applied only once to replace the existential sentence.

iii. This rule states that one can infer $P(c)$ from the formula given in the form of $\exists \times P(x)$ for a new constant symbol c.

iv. The restriction with this rule is that c used in the rule must be a new term for which $P(c)$ is true.

v. It can be represented as :

$$= \frac{\exists \times P(x)}{P(c)}$$

4. Existential introduction :

i. An existential introduction is also known as an existential generalization, which is a valid inference rule in first order logic.

ii. This rule states that if there is some element c in the universe of discourse which has a property P, then we can infer that there exists something in the universe which has the property P.

iii. It can be represented as :

$$= \frac{P(c)}{\exists \times P(x)}$$

Que 3.3. What are the properties of first order logic ?

Answer

Properties of first order logic :

1. It has ability to represent facts about some or all of the objects in the universe.

2. It enables to represent law and rules extracted from real world.
3. It is useful language representation in mathematics, philosophy and AI related fields.
4. It represents facts in more realistic manner rather than just the true or false statement.

Que 3.4. **Translate the following sentences into formulas in predicate logic and clausal form :**

i. John likes all kind of food.
ii. Apples are food.
iii. Chicken is food.
iv. Anything any one eats and is not killed by is food.
v. Bill eats peanuts and is still alive.
vi. Sue eats everything Bill eats. **AKTU 2019-20, Marks 07**

Answer

Predicate logic :

i. $\forall$ x food (x) $\rightarrow$ likes (John, x)
ii. food (apples)
iii. food (chicken)
iv. $\forall$ x $\forall$ y Eats (x, y) $\wedge \neg$ killed (y) $\rightarrow$ food (x)
v. Eats (Peanuts, Bill) $\wedge$ alive (Bill)
vi. $\forall$ x, Eats (x, Bill) $\rightarrow$ Eats (x, sue)

Clause form :

i. $\neg$ food (x) $\vee$ likes (John, x)
ii. food (apple)
iii. food (chicken)
iv. $\neg$ Eats (x, y) $\vee$ killed (y) $\vee$ food (x)
v.
   a. Eats (Peanut, Bill)
   b. Alive (Bill)
vi. $\neg$ Eats (x, Bill) $\vee$ Eats (x, sue)

Que 3.5. **Translate following sentences in formulas in predicate logic and casual form :**

a. Mutton is food.
b. Anything one eats and it does not kill is a food.
c. Rajiv eats everything that Sue eats.
d. Kin eats peanuts and is still alive.
e. John will marry Mary if Mary loves John.

AKTU 2015-16, Marks 10

Answer

Predicate logic :

a. Food (Mutton)

b. $\forall x \ \forall y$ Eats $(x, y) \wedge \neg$ killed $(y) \rightarrow$ food (x)

c. Eats (sue, x) $\rightarrow$ eats (Rajiv, x)

d. Eats (peanuts, kin) $\rightarrow$ alive (kin)

e. $\forall x \ \forall y$ Marry $(x, y) \rightarrow$ loves (y, x)

Casual form :

a. Food (Mutton)

b. $\neg$ (Eats $(x, y) \wedge$ killed $(y) \vee$ food (x)

c. $\neg$ Eats $(x$, Sue) $\vee$ Eats $(x$, Rajiv)

d. i. Eats (Peanut, kin)

ii. Alive (kin)

e. Marry (John, Mary) $\rightarrow$ loves (Mary, John)

Que 3.6. Define a well-formed formula (WFF) and list some of the rules of inference. **AKTU 2020-21, Marks 07**

Answer

1. Well formed formula - Propositional logic uses a symbolic language to represent the logical structure, or form, of a compound proposition.
2. The symbolic language has rules of syntax, grammatical rules for putting symbols together in the right way.
3. Any expression that obeys the syntactic rules of propositional logic is called a well-formed formula or WFF.

Rules of inference :

1. **Modus Ponens :** The Modus Ponens rule states that if P and $P \rightarrow Q$ is true, then we can infer that Q will be true.

Example :

Statement-1 : "If I am sleepy then I go to bed". $(P \rightarrow Q)$

Statement-2 : "I am sleepy". (P)

Conclusion : "I go to bed". (Q)

Hence, we can say that, if $P \rightarrow Q$ is true and P is true then Q will be true.

Proof by truth table :

P	Q	$P \rightarrow Q$
0	0	0
0	1	1
1	0	0
1	1	1 ←

2. **Modus Tollens :** The Modus Tollens rule state that if $P \rightarrow Q$ is true and Q is true, then $\neg P$ will also true.

Example :

Statement-1 : "If I am sleepy then I go to bed". $(P \rightarrow Q)$

Statement-2 : "I do not go to the bed". $(\sim Q)$

Statement-3 : Which infers that "I am not sleepy". $(\sim P)$

Proof by truth table :

P	Q	$\sim P$	$\sim Q$	$P \rightarrow Q$
0	0	1	1	1 ←
0	1	1	0	1
1	0	0	1	0
1	1	0	1	1

3. **Hypothetical Syllogism :** The Hypothetical Syllogism rule state that if $P \rightarrow R$ is true whenever $P \rightarrow Q$ is true and $Q \rightarrow R$ is true.

Example :

Statement-1 : If you have my home key then you can unlock my home. $(P \rightarrow Q)$

Statement-2 : If you can unlock my home then you can take my money. $(Q \rightarrow R)$

Conclusion : If you have my home key then you can take my money. $(P \rightarrow R)$

Proof by truth table :

P	Q	R	$P \rightarrow Q$	$Q \rightarrow R$	$P \rightarrow R$
0	0	0	1	1	1 ←
0	0	1	1	1	1
0	1	0	1	0	1
0	1	1	1	1	1 ←
1	0	0	0	1	1
1	0	1	0	1	1
1	1	0	1	0	0
1	1	1	1	1	1 ←

4. **Disjunctive Syllogism :** The Disjunctive Syllogism rule state that if $P \vee Q$ is true, and $\neg P$ is true, then Q will be true.

Example :

Statement-1 : Today is Sunday or Monday. $(P \wedge Q)$

Statement-2 : Today is not Sunday. $(\neg P)$

Conclusion : Today is Monday. (Q)

Proof by truth-table :

P	Q	$\neg P$	$P \vee Q$
0	0	1	0
0	1	1	1 ←
1	0	0	1
1	1	0	1

5. **Addition :** The Addition rule states that If P is true, then $P \vee Q$ will be true.

Example :

Statement-1 : I have a vanilla ice-cream. (P)

Statement-2 : I have Chocolate ice-cream. (Q)

Conclusion : I have vanilla or chocolate ice-cream. ($P \vee Q$)

Proof by truth-table :

P	Q	$P \vee Q$
0	0	0
1	1	1 ←
0	1	1
1	1	1 ←

6. **Simplification :** The simplification rule state that if $P \wedge Q$ is true, then Q or P will also be true.

Proof by truth-table :

P	Q	$P \wedge Q$
0	0	0
1	0	0 ←
0	1	0
1	1	1 ←

7. **Resolution :** The resolution rule state that if $P \vee Q$ and $\neg P \wedge R$ is true, then $Q \vee R$ will also be true.

Proof by truth-table :

P	$\neg P$	Q	R	$P \vee Q$	$\neg P \wedge R$	$Q \vee R$
0	1	0	0	0	0	0
0	1	0	1	0	0	1
0	1	1	0	1	1	1 ←
0	1	1	1	1	1	1 ←
1	0	0	0	1	0	0
1	0	0	1	1	0	1
1	0	1	0	1	0	1
1	0	1	1	1	0	1 ←

Que 3.7. **Explain the conversion procedure of given formula into normal form.** **AKTU 2017-18, Marks 10**

Answer

The steps used to convert a given formula into its normal form are as follows :

Step 1 : Eliminate implications and biconditionals. For this, use the laws :

$$(A \rightarrow B) = \sim A \vee B$$

$$(A \leftrightarrow B) = (A \rightarrow B) \wedge (B \rightarrow A)$$

$$= (\sim A \vee B) \wedge (\sim B \vee A)$$

Step 2 : Reduce the NOT symbol by the formula $(\sim(\sim A)) = A$ and apply De Morgan's theorem to bring negations before the atoms.

$$\sim(A \vee B) = \sim A \wedge \sim B$$

$$\sim(A \wedge B) = \sim A \vee \sim B$$

Step 3 : Use distributive laws to obtain the normal form.

$$A \wedge (B \vee C) = (A \wedge B) \vee (A \wedge C)$$

$$A \vee (B \wedge C) = (A \vee B) \wedge (A \vee C)$$

Que 3.8. **Prove that following sentence is valid :**

"If prices fall then sell increases. If sell increases then John makes the whole money. But John doesn't make the whole money. Therefore, prices do not fall." **AKTU 2017-18, Marks 10**

Answer

1. If prices fall then sell increases.
   The logical form of the sentence is : If P then Q.
2. If sell increases then John makes the whole money.
   The logical form of the sentence is : If Q then R.
3. We have to check the validity of the sentence.
   John doesn't make the whole money. Therefore, Prices do not fall.

$$\neg R \rightarrow \neg P$$

Hence, from (1) and (2) and by using Modus Ponens rule we get :

$P \rightarrow Q$ (Price falls $\rightarrow$ Sell increases)

$Q \rightarrow R$ (Sell increases $\rightarrow$ John makes whole money)

$P \rightarrow R$ (Price falls $\rightarrow$ John makes whole money)

From transposition rule,

$$P \rightarrow R \text{ infer } \neg R \rightarrow \neg P$$

We conclude that John does not make the whole money, therefore prices do not fall.

Hence, the given argument is valid.

PART-2

Prolog Programming.

Questions-Answers

Long Answer Type and Medium Answer Type Questions

Que 3.9. Define Prolog programming. What are the features of Prolog programming ?

Answer

1. Prolog is known for its in-built depth first search engine and for quick prototyping.
2. A Prolog program consists of a database of facts and rules. There is no structure imposed on a Prolog program, there is no main procedure and there is no nesting of definitions.
3. All facts and rules are global in scope and the scope of a variable is the fact or rule in which it appears.
4. The readability of a Prolog program is left up to the programmer.

The main features of Prolog are :

1. **Rule-based programming :** The rule-based programming allows the program code to be written in the form which is more declarative than procedural.
2. **Built-in pattern matching :** It has an important feature of built-in pattern matching.
3. **Backtracking execution :** Backtracking provides the means for the flow of control in the program.

Que 3.10. Explain the following terms with reference to Prolog programming :

i. Clauses
ii. Predicates
iii. Domains

Answer

i. Clauses :

1. Clauses are the structural elements of a program. A Prolog programmer develops a program by writing a collection of clauses in a text file.
2. The programmer then uses the consult command, specifying the name of the text file, to load the clauses into the Prolog environment.
3. Following are two types of clauses :

a. Facts : A fact is an atom or structure followed by a full stop. Examples of valid Prolog syntax for defining facts are : cold., male(amit)., and father(amit, mohan)..

b. Rules : A rule consists of a head and a body. The head and body are separated by a :- and followed by full stop. If the body of a clause is true then the head of the clause is true. Examples of valid Prolog syntax for defining rules are : bigger(X,Y) :- X > Y. and parents(F,M,C) :- father(F,C), mother(M,C).

ii. Predicates :

1. Each predicate has a name, and zero or more arguments. The predicate name is a Prolog atom.
2. Each argument is an arbitrary Prolog term.
3. A predicate with name Pred and *N* arguments is denoted by Pred/*N*, which is called a predicate indicator. *N* is called the arity of the predicate.
4. A predicate is defined by a collection of clauses.
5. A clause is either a rule or a fact. The clauses that constitute a predicate denote logical alternatives : If any clause is true, then the whole predicate is true.

iii. Domains : The arguments to the predicates must belong to known Prolog domains. A domain can be a standard domain, or it can be one you declare in the domains section.

Examples :

If you declare a predicate my_predicate(symbol, integer) in the predicates section, like this :

PREDICATES

my_predicate(symbol, integer)

you don't need to declare its argument's domains in a domains section, because symbol and integer are standard domains. But if you declare a predicate my_predicate(name, number) in the predicates section, like this :

PREDICATES

my_predicate(name, number)

you will need to declare suitable domains for name and number. Assuming you want these to be symbol and integer respectively, the domain declaration looks like this :

DOMAINS

name = symbol

number = integer

PREDICATES

my_predicate(name, number)

PART-3

Unification.

Questions-Answers

Long Answer Type and Medium Answer Type Questions

Que 3.11. What do you mean by unification ?

Answer

1. Unification is the process of finding substitutions for lifted inference rules, which can make different logical expression to look similar (identical).
2. Unification is a procedure for determining substitutions needed to make two first order logic expressions match.
3. Unification is important component of all first order logic inference algorithms.
4. The unification algorithm takes two sentences and returns a unifier for them, if one exists.

Que 3.12. Explain unification algorithm used for reasoning under predicate logic with an example. **AKTU 2016-17, Marks 10**

Answer

1. In order to do resolution for expressions in the predicate logic, we use unification algorithm to locate pairs of literals that cancel out.
2. We need to use the unifier produced by the unification algorithm to generate the resolvent clause.

For example, we want to resolve following two clauses :

a. man (Marcus)

b. $\neg$ Man (x) $\vee$ mortal (x)

3. The literal man (Marcus) can be unified with the literal man (x) with the substitution Marcus / x, telling us that for x = Marcus, $\neg$ man (Marcus) is false.

4. Now we cannot simply cancel out the two man literals as in case of propositional logic and generate logic we now conclude only that mortal (Marcus) must be true which we get by applying the result of the unification process to the resolvent.

5. The resolution process can then proceed to discover whether mortal (Marcus) leads to a contradiction with other available clauses.

Unification algorithm : Verify (U, V)

1. If U and V are both variables or constants then

   a. If U and V are identical, then return null.

   b. If U is a variable then if U occurs in V, then return {FAIL} else return {U/V}

   c. If V is a variable then if V occurs in U, return {FAIL} else return {U/V}.

   d. Return {FAIL}.

2. If the initial predicate symbols in U and V are not identical, return {FAIL}.

3. If U and V have a different number of arguments, then return {FAIL}.

4. Set SUBSET to NULL.

5. For $i \leftarrow l$ to the number of arguments of U.

   a. Call unify with the i^{th} argument of U and the i^{th} argument of V putting the result in S.

   b. If S contains {FAIL}, return {FAIL}.

   c. If S is not equal to NULL

      i. Apply step 5 to be the remainder of both U and V.

      ii. Set subset equal to APPENDS (S, SUBSET).

6. Return SUBSET.

Que 3.13. Trace the operation of the unification algorithm on each of the following pairs of literals :

i. f(Marcus) and f(Caesar)

ii. f(x) and f(g(y))

iii. f(Marcus, g(x, y)) and f(x, g(Caesar, Marcus))

AKTU 2020-21, Marks 07

Answer

a. Here unification attempt is between

f(Marcus) and f(Caesar)

This returns FAIL.

b.

i. We want to unify the expression f(x) and f(g(y)).

ii. The two instances of ' f ' match fine.

iii. Next we compare x and g(y), and decide that if we substitute g(y) for x, they could match.

iv. We will write that substitution as (g(y)/x).

c. Unifying the first pair of arguments produces : (Marcus/x). If we make that substitution, then the next unification attempt is between

g(Marcus. y) and g(Caesar. Marcus)

This returns FAIL.

PART-4

Forward Chaining, Backward Chaining.

Questions-Answers

Long Answer Type and Medium Answer Type Questions

Que 3.14. Elaborate forward and backward chaining.

AKTU 2016-17, Marks 05

Answer

Forward chaining :

1. Forward chaining is a method of reasoning when using inference rules in artificial intelligence.
2. Forward chaining starts with the available data and uses inference rules to extract more data (from an end user) until an optimal goal is reached.
3. An inference engine using forward chaining searches the inference rules until it finds one where the If clause is known to be true.

4. When found it can conclude, or infer, the Then clause, resulting in the addition of new information to its dataset.
5. Inference engines will often cycle through this process until an optimal goal is reached.
6. For example, suppose that the goal is to conclude the colour of my pet Bruno given that he croaks and eats flies, and that the rule base contains the following two rules :
   a. If *X* croaks and eats flies - Then *X* is a frog.
   b. If *X* is a frog - Then *X* is red.

Backward chaining :

1. Backward chaining starts with a list of goals (or a hypothesis) and works backwards to see if there is data available that will support any of these goals.
2. An inference engine use backward chaining would search the inference rules until it finds one which has a Then clause that matches a desired goal.
3. If the If clause of that inference rule is not known to be true, then it is added to the list of goals.
4. For example, suppose that the goal is to conclude the colour of my pet Bruno given that he croaks and eats flies, and that the rulebase contains the following two rules :
   a. If *X* croaks and eats flies - Then *X* is a frog.
   b. If *X* is a frog - Then *X* is red.

Que 3.15. Write down the properties of forward chaining and backward chaining.

Answer

Properties of forward chaining :

1. It is a down-up approach, as it moves from bottom to top.
2. It is a process of making a conclusion based on known facts or data, by starting from the initial state and reaches the goal state.
3. Forward chaining approach is also called as data-driven as we reach to the goal using available data.
4. Forward chaining approach is commonly used in the expert system, such as business, and production rule systems.

Properties of backward chaining :

1. It is known as a top-down approach.
2. Backward chaining is based on Modus Ponens inference rule.
3. In backward chaining, the goal is broken into sub-goal or sub-goals to prove the facts true.

4. It is called a goal-driven approach, as a list of goals decides which rules are selected and used.
5. Backward chaining algorithm is used in game theory, automated theorem proving tools, inference engines, proof assistants, and various AI applications.
6. The backward chaining method mostly used a depth-first search strategy for proof.

Que 3.16. Differentiate between forward and backward chaining of inference with the help of an example.

AKTU 2018-19, Marks 10

AKTU 2019-20, Marks 07

Answer

S. No.	Forward chaining	Backward chaining
1.	Forward chaining is a data driven method.	Backward chaining is a goal driven method.
2.	It uses planning, monitoring and controlling method.	It uses diagnosis method.
3.	It uses bottom-up processing.	It uses top-down processing.
4.	Forward chaining finds possible conclusions supported by given facts.	Backward chaining finds facts that support a given hypothesis.
5.	Forward chaining is similar to breadth-first search.	Backward chaining is similar to depth-first search.
6.	For example : CLIPS.	For example : PROLOG.

PART-5

Resolution.

Questions-Answers

Long Answer Type and Medium Answer Type Questions

Que 3.17. **Write a short note on resolution.**

OR

What is resolution ? Discuss the role of resolution in predicate logic. **AKTU 2016-17, Marks 10**

Answer

1. Resolution is a proof, procedure that carries out a single operation, the variety or processes involved in reasoning with statements in predicate logic.
2. Resolution operates on statements that have not been converted into a very convenient form.
3. Resolution procedure is proved by reputation, in other words, to prove a statement resolution attempts to show that negation of statement produces a contradiction with the known fact, *i.e.*, it is unsatisfiable.

Role of resolution in predicate logic :

1. Resolution is an inference step required to build a complete inference system for predicate logic in clause form.
2. Applying the rule of resolution to the clauses makes the resolution inference highly suitable for computer implementation.

Que 3.18. **Discuss resolution in propositional and predicate logic.**

Answer

Resolution of propositional logic : In resolution of propositional logic, the procedure for producing a proof by resolution of propositional S with respect to a set of axioms F is as follows :

1. Convert all the propositions of F to clause form.
2. Negate S and convert the result to clause form. Add it to the set of clauses obtained in step 1.
3. Repeat until either a contradiction is found or no progress can be made :
   i. Select two clauses, call these parent clauses.
   ii. Resolve them together. The resulting clause called the resolvent will be the disjunction of all of the literals of both of the parent clauses with the following exception : If there are any pairs of literals L and $\sim L$, such that one of the parent clause contains L and other $\sim L$, then select one such pair and eliminate both L and $\sim L$ from the resolvent.
   iii. If the resolvent is the empty clause, then a contradiction has been found. If it is not, then add it to the set of clauses available to the procedure.

Resolution in predicate logic :

1. Convert all the statements of F to clause form.
2. Negate S and convert the result to clause form. Add it to the set of clauses obtained in step 1.
3. Repeat until either a contradiction is found, no progress can be made or a predetermined amount of effort has been expended :
   i. Select two clauses, call these parent clauses.
   ii. Resolve them together. The resolvent will be disjunction of all literals of both of parent clause with appropriate substitutions performed and with the following exception : If there is one pair of literals T_1 and ~ T_2 such that one of the parent clauses contains T_1 and the other contains T_2 and if T_1 and T_2 are unifiable, then neither T_1 nor T_2 should appear in the resolvent.
   iii. If the resolvent is the empty clause, then a contradiction has been found. If it is not, then add it to the set of clauses available to the procedure.

Que 3.19. Explain resolution in predicate logic with suitable example. **AKTU 2020-21, Marks 07**

Answer

Resolution in predicate logic :

1. Convert all the statements of F to clause form.
2. Negate S and convert the result to clause form. Add it to the set of clauses obtained in step 1.
3. Repeat until either a contradiction is found, no progress can be made or a predetermined amount of effort has been expended :
   i. Select two clauses, call these parent clauses.
   ii. Resolve them together. The resolvent will be disjunction of all literals of both of parent clause with appropriate substitutions performed and with the following exception : If there is one pair of literals T_1 and ~ T_2 such that one of the parent clauses contains T_1 and the other contains T_2 and if T_1 and T_2 are unifiable, then neither T_1 nor T_2 should appear in the resolvent.
   iii. If the resolvent is the empty clause, then a contradiction has been found. If it is not, then add it to the set of clauses available to the procedure.

PART-6

Knowledge Representation.

Questions-Answers

Long Answer Type and Medium Answer Type Questions

Que 3.20. Briefly describe the meaning of knowledge representation and knowledge acquisition. What procedure is followed for knowledge acquisition ? Explain.

Answer

Knowledge representation :

1. Knowledge representation is the study of ways of picturization of knowledge and how effectively it resembles the representation of knowledge in human brain.
2. Knowledge representation has two entities :
   a. **Facts :** Facts are the truth in some relevant world.
   b. **Representation :** Representation is the presentation of facts in some chosen formalism.
3. For example :

   Fact : Charlie is a dog.

   Representation of fact using mathematical logic : Dog (Charlie)
4. Knowledge representation should possess following characteristics :
   a. Representation scheme should have a set of well defined syntax and semantic.
   b. It should have good expressive capacity.
   c. It must be effective.

Knowledge acquisition :

1. Knowledge acquisition is the process of acquiring knowledge from a human expert for an expert system, which must be carefully organized into IF-THEN rules or some other form of knowledge representation.
2. Knowledge acquisition is the process of absorbing and storing new information in memory, the success of which is determined by how well the information can later be retrieved from memory.

3. The process of storing and retrieving information depends heavily on the representation and organization of the information.

Procedure for knowledge acquisition :

1. **Identification :** Break the problems into parts.
2. **Conceptualisation :** Identify the concepts.
3. **Formalisation :** Represent the knowledge.
4. **Implementation :** Programming.
5. **Testing :** Validate of knowledge.

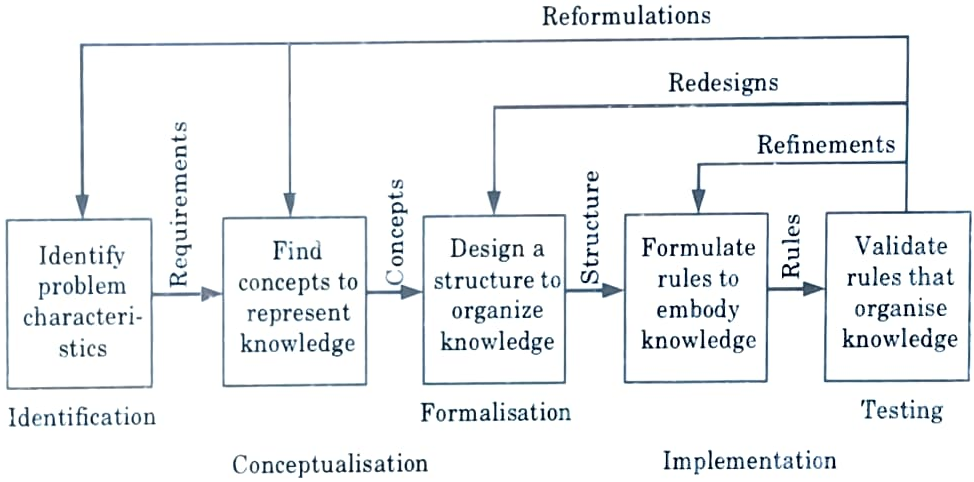


Fig. 3.20.1. Stages of knowledge acquisition.

Que 3.21. Describe the techniques of knowledge representation.

Answer

Techniques of knowledge representation :

1. **Logical representation :**
   i. Logical representation is a language with some concrete rules which deals with propositions and has no ambiguity in representation.
   ii. Logical representation means drawing a conclusion based on various conditions.
   iii. It consists of precisely defined syntax and semantics which supports the sound inference.
   iv. Each sentence can be translated into logics using syntax and semantics.
2. **Semantic network representation :**
   i. Semantic networks are alternative of predicate logic for knowledge representation.

ii. In semantic networks, we can represent our knowledge in the form of graphical networks.

iii. This network consists of nodes representing objects and arcs which describe the relationship between those objects.

iv. Semantic networks can categorize the object in different forms and can also link those objects.

3. **Frame representation :**

i. A frame is a record like structure which consists of a collection of attributes and its values to describe an entity in the world.

ii. Frames are the AI data structure which divides knowledge into substructures by representing stereotypes situations.

iii. 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.

4. **Production rules :**

i. Production rules system consist of (condition, action) pairs which mean, "If condition then action".

ii. In production rules agent checks for the condition and if the condition exists then production rule and corresponding action is carried out.

iii. The condition part of the rule determines which rule may be applied to a problem. And the action part carries out the associated problem solving steps. This complete process is called a recognize-act cycle.

Que 3.22. What are the desirable properties of good knowledge representation schemes ? **AKTU 2015-16, Marks 10**

Answer

A good system for the representation of knowledge in a particular domain should possess the following four properties :

1. **Representational adequacy :** The ability to represent all kinds of knowledge that are needed in that domain.
2. **Inferential adequacy :** The ability to manipulate the representational structures in such a way as to derive new structures corresponding to new knowledge inferred from old.
3. **Inferential efficiency :** The ability to incorporate into the knowledge structure, additional information that can be used to focus the attention of the inference mechanisms in the most promising directions.
4. **Acquisitional efficiency :** The ability to acquire new information easily. The simplest case involves direct insertion, by a person, of new knowledge into the database. Ideally, the program itself would be able to control knowledge acquisition.

PART-7

Ontological Engineering, Categories and Objects.

Questions-Answers

Long Answer Type and Medium Answer Type Questions

Que 3.23. What is ontological engineering ? Explain the concept of categories and objects.

Answer

A. Ontological engineering :

1. The process of representing the abstract concepts like actions, time which are related to real world domains is defined as Ontological Engineering.
2. This process is tough and lengthy because in real world, objects have many different characteristics with various values.
3. These values differ over time.

B. The concept of categories and objects :

1. The classification of objects can be done in to several categories.
2. Doing categorization helps in knowledge representation.
3. In real world interaction is with the individual object but it really helps to deal with the object at the level of categories.
4. Due to categorization searching would be fast because objects are well organized.
5. **For example :** A boy wants to buy a special football such as FB7.
6. Categories help to make predictions about objects once they are classified.
7. Agent can infer presence of certain objects from perceptual input, and then decide category on the basis of perceived properties of the objects.

Que 3.24. How do we represent categories in first order logic ?

Answer

We can represent categories in first order logic using either of the two ways :

1. **Predicates :** We can use predicate to categorize object.

   For example : Member (f, football)

   Here Member is predicate and 'f' is a object, categorization under 'Member'.

2. **Objects :** We can use object itself to categorized the set of objects.
   For example : Football (f)
   Here 'f' is object categorized as Football.

PART-8

Events, Mental Events and Mental Objects.

Questions-Answers

Long Answer Type and Medium Answer Type Questions

Que 3.25. Explain mental events and mental objects.

Answer

Mental Events and Mental Objects :

1. AI agent should be able to deduce beliefs (facts which are true) from current knowledge it has. Agent should possess knowledge about beliefs or about deducing the beliefs.
2. For single-agent environment, knowledge about one's own knowledge and the reasoning process is useful for controlling inference.

   For example :

   Agent wants to buy a pencil. As per agent's knowledge agent will reach to a stationary shop to purchase pencil.
3. In multi-agent environment, the knowledge and reasoning process becomes important to reason about mental state of the other agent. To buy a 'pencil' of certain brand (where the pencil is ?) agent should enquire it to shop owner (another agent). This process will happen only if agent has the knowledge that "for various unknown brands I should ask the shopkeeper (because he has knowledge about it)."
4. One's own knowledge help to construct future plans that helps to take relevant action.
5. To make belief, knowledge and reasoning theory to work, we need to model the mental objects that are in someone's head (that is something's knowledge base) and the mental processes that manipulate them.

PART-9

Reasoning Systems for Categories, Reasoning with Default Information.

Questions-Answers

Long Answer Type and Medium Answer Type Questions

Que 3.26. What are various reasoning systems for organizing categories ?

Answer

For organizing categories there are two types of reasoning systems :

A. Semantic networks :

1. It provides graphical aids for visualizing a knowledgebase and efficient algorithms for inferring properties of an object on the basis of its category membership.
2. Semantic network is a graph which has nodes representing objects and their categories, and arc representing relationships between objects.
3. A typical graphical notation displays object or category names in oval or boxes and connects them with the labeled arcs.

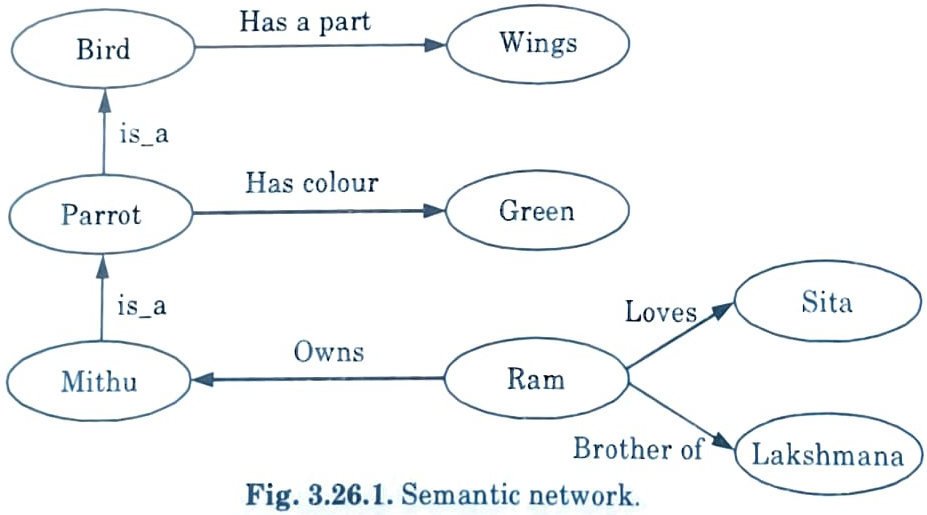


Fig. 3.26.1. Semantic network.

4. Semantic network provide direct indexing for objects, categories and the links between them.
5. Semantic network can represent default values for categories.

B. Description logic :

1. These are the notations that are designed to describe definitions and properties of categories.
2. Description logic system formalize information represented by the semantic network.
3. The inference techniques used in description logic are :

i. **Subsumption :** Checking if one category is a subset of another by comparing their definitions.

ii. **Classification :** Checking whether an object belongs to a category.

iii. **Consistency :** It is used for category definition to check if membership criteria are logically satisfiable.

4. Generally a problem instance is solved by describing it and then asking if it is subsumed by one of several possible solution categories.
5. The statements in description logic do not contain CLASSIC language saying.

Que 3.27. Explain default reasoning.

Answer

Default reasoning :

1. There are two approaches (both are logic type) to Default reasoning.
2. One is Non-monotonic logic and the other is Default logic.

A. Non-Monotonic logic :

1. Non-monotonic logic says, "the truth of a proposition may change when new information (axioms) are added and a logic may be build to allow the statement to be retracted."
2. Non-monotonic logic is predicate logic with one extension called Modal operator, M which means "consistent with everything we know". The purpose of M is to allow consistency.
3. A way to define consistency with PROLOG notation is :

   To show that fact P is true, we attempt to prove $\neg P$.

   If we fail we may say that P is consistent since

   $\neg P$ is false.

   For example :

   $\forall \times$ plays_instrument $(\times) \wedge$ Manage $(\times) \rightarrow$ jazz_musician $(\times)$.

   States that for all $\times$, the $\times$ plays an instrument and if the fact that $\times$ can manage is consistent with all other knowledge then we can conclude that $\times$ is a jazz musician.

B. Default logic :

1. Default logic initiates a new inference rule :

   $\frac{A : B}{C}$ where

   A is known as the prerequisite.

   B is the justification, and

   C as the consequent.

2. The above inference rule is read as :

   "If A, and if it is consistent with the rest of what is known to assume that B. Then conclude that C."

3. The rule says that given the prerequisite, the consequent can be inferred, provided it is consistent with the rest of the data.

4. The idea behind non-monotonic reasoning is to reason with first order logic, and if an inference can not be obtained then use the set of default rules available within the first order formulation.

4

UNIT

Software Agents

CONTENTS

PART-1

Architecture for Intelligent Agents.

Questions-Answers

Long Answer Type and Medium Answer Type Questions

Que 4.1. What is an intelligent agent ?

Answer

1. An intelligent agent is an AI agent that is capable of flexible autonomous action in order to meet its design objectives, where flexibility comprises of three aspects namely,
   i. **Reactivity :** Intelligent agents are able to perceive their environment, and respond in a timely fashion to changes that occur in it in order to satisfy their prior designed objectives.
   ii. **Pro-activeness :** Intelligent agents are able to exhibit goal-directed behavior by taking the initiative in order to satisfy their prior designed objectives.
   iii. **Social ability :** Intelligent agents are capable of interacting with other agents in order to satisfy their prior design objectives.
2. All above 3 properties are highly demanding and are critical and complex to implement.
3. This complexity arises from several factors like intelligent agent's interaction with environment, its prior knowledge, its hardware capabilities, etc.

Que 4.2. What are the categories of agents on the basis of architecture ?

Answer

Categories of agents on the basis of architecture are as follows :

1. **Logic based agents :** These agents carries out decision making through logical deductions.
2. **Reactive agents :** These agents carries out decision making through procedure implementation in some form of direct mapping from situation to action.

3. **Belief-desire-intention agents :** These agents carries out decision making that depends upon the manipulation of data structures which are used to represent the agent's beliefs, agent's desires, and agent's intentions.
4. **Layered architectures :** Here the decision making is done through various software layers, each of which explicitly reasons about the environment at different levels of abstraction as per the requirement of problem under consideration.

Que 4.3. Explain logic based agent architecture with their advantages and disadvantages.

Answer

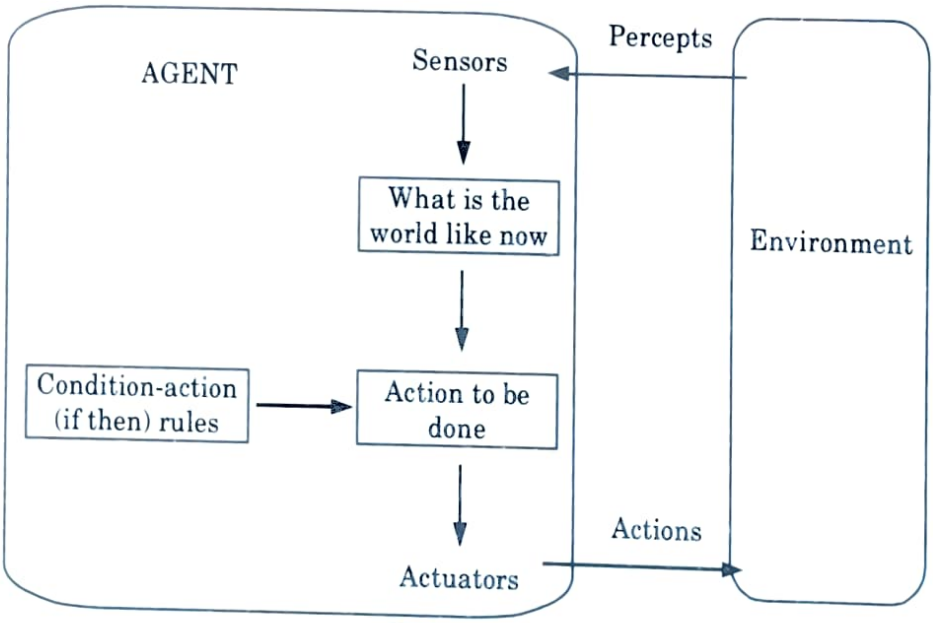


Fig. 4.3.1.

1. It is the traditional approach to building artificially intelligent system, (knows as symbolic AI) suggest that intelligent behavior can be generated in a system by providing that system a symbolic representation of its environment and its desired behavior, and syntactically manipulating this representation.
2. In these systems, the symbolic representations are logical formulae, and the syntactic manipulation corresponds to logical deduction, or theorem proving.
3. The idea of agents as theorem proven is highly attractive.
4. The traditional approach for the implementation of a system that satisfy this specification would involve refining the specification through a series of progressively more concrete stages, until finally an implementation is completely done.

Logic based approach advantages :

1. In logic-based approaches to building agents, decision making is viewed as deduction.
2. Logic-based approaches are good to work with and have a clean (logical) semantic due to which they can be used over long period of time.

Logic based approach disadvantages :

1. The inbuilt computational complexity of theorem proving makes it questionable whether agents as theorem provers can operate effectively in time-constrained environments.
2. Decision making in such agents is predicated on the assumption of calculative rationality.
3. The issues associated with representing and reasoning about complex, dynamic, possibly physical environments are also essentially unsolved that makes developing logic agent a tedious task.

Que 4.4. Explain reactive agent architecture with their advantages and disadvantages.

Answer

1. The reactive agent approaches are sometime referred to as behavioral, situated, and reactive.
2. It is referred to as behavioral because a common theme is that of developing and combining individual behaviors.
3. It is referred to as situated because a common theme is that of agents actually situated in some environment, rather than being disembodied from it.
4. It is referred to as reactive because such systems are 100% understood to be just reacting to an environment, without reasoning about it.
5. The reactive agent architecture is subsumption architecture, which is comparatively the best-known reactive agent architecture

Reactive agent approach advantages :

1. The overall time complexity of the subsumption action function is not worse than $O(n^2)$, where n is the larger of the number of behaviors or number of percepts.
2. The major advantages of reactive approaches are simplicity, economy, computational tractability, robustness against failure.

Reactive agent approach disadvantages :

1. In reactive agent, if the agents do not employ models of their environment, then they must have sufficient information available in their local environment for them to determine an acceptable action.

2. It is difficult model, for purely reactive agents that can be designed to learn from experience so as to improve their performance over time.

Que 4.5. Explain belief desire intention agent architecture with advantages and disadvantages.

Answer

Belief-Desire-Intention (BDI) Agent Architecture :

1. This architecture has thought of from philosophical tradition of understanding practical reasoning, the process of deciding, moment by moment, which action to perform in the furtherance of one's goals.
2. The terms beliefs, desires and intentions can be defined as follows :
   - i. **Beliefs :** Beliefs represent the informational state of the agent, that is agent's beliefs about the world (including itself and other agents).
   - ii. **Desires :** Desires represent the motivational state of the agent.
   - iii. **Goals :** A goal is an ultimate desire that has been adopted for active pursuit by the agent.
   - iv. **Intentions :** Intentions represent the deliberative state of the agent, indicating what the agent has chosen to do.
   - v. **Plans :** The sequences of action (recipes or knowledge areas) that an agent can perform to achieve one or more of its intentions are termed as plans.
   - vi. **Events :** Events are triggers for reactive activity to be carried out by the agent. An event may update beliefs, trigger plans or modify goals.
3. In this architecture practical reasoning involves two crucial processes, namely to decide what goals are to achieve and how these goals are going to get achieved.

BDI approach - advantages :

1. As this model uses a standard human reasoning process to reach to goal it is easy to understand.
2. It has clear functional decomposition, which indicates what sorts of subsystems might be required to build an agent.

BDI approach - disadvantages :

1. Main difficulty lies in knowing how to efficiently implement all BDI model functions.

Que 4.6. Explain layered architecture.

Answer

Layered architecture :

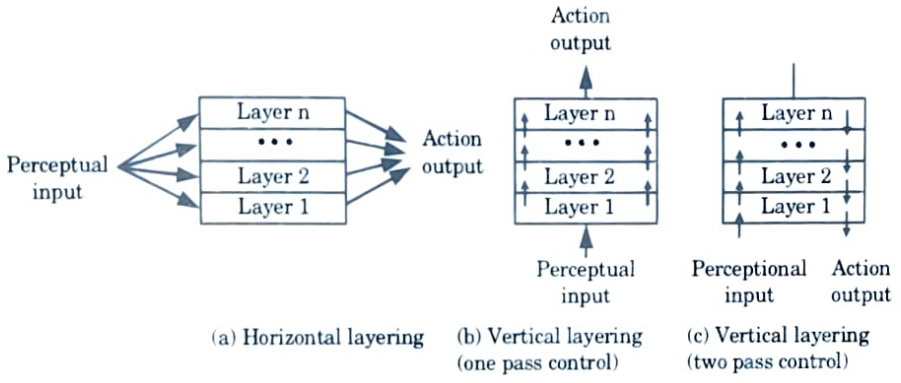


Fig. 4.6.1. Layered agent architecture types.

1. In layered architecture various subsystems are arranged into a hierarchy of interacting layers.
2. Formally, there are following two types of control flow within layered architectures :
   a. **Horizontal layering :** In horizontally layered architectures, the software layers are each directly connected to the sensory input and action output. In effect, each layer itself acts like an agent, generates what action to perform.
   b. **Vertical layering :** In vertically layered architectures, sensory input and action output are each dealt with by at most one layer each.

PART-2

Agent Communication.

Questions-Answers

Long Answer Type and Medium Answer Type Questions

Que 4.7. Explain multiagent system and agent communication.

Answer

Multiagent Systems and Agent's Communication :

1. While acting in real world, an agent may not be always in a singleton mode.

2. It has to deal with the situation wherein other agents and related factors are affecting the agent's environment and in turn agent's behavior.
3. Such systems wherein multiple agent work together, communicate, co-operate and deal with the situation are termed as multi-agent system.
4. Multi-agent system are essentially distributed systems which more efficient in the sense that they can be optimized and are mostly easier to understand and easier to develop, especially when the problem being solved is itself distributed.
5. Distribution can lead to computational algorithms that might not have been discovered with a centralized approach.
6. The data and information itself is distributed spanning at different geographical locations and needs to be handled through multiple agents.
7. Data can come from various domains and multiple devices or components are involved in data generation.
8. The system itself is too big and complex that needs to be separated in multiple components so as to reduce its complexity and size that can be handled easily.

Que 4.8. Write down the characteristics of multiagent system.

Answer

Characteristics of Multi-Agent Systems :

1. Each agent has just incomplete information and is restricted in its capabilities.
2. The system control is distributed.
3. Data is decentralized.
4. Computation is asynchronous.
5. Multi-agent environments are typically open and have no centralized design
6. Multi-agent environments have agents that are autonomous and distributed, and may be self-interested or cooperative.

PART-3

Negotiation and Bargaining.

Questions-Answers

Long Answer Type and Medium Answer Type Questions

Que 4.9. Explain negotiation and bargaining.

Answer

1. In a multi-agent system negotiation is form of interaction that occurs among agents with different goals.
2. Major challenge of negotiation and bargaining is to allocate scarce resources among agents representing self-interested parties. The resources can be bandwidth, commodities, money, processing power etc. The resource becomes scarce as competing claims for it can't be simultaneously satisfied.
3. Negotiation and bargaining is a process by which a joint decision is reached by two or more agents, each trying to reach an individual goal or objective.
4. The major features of negotiation and bargaining are :
   i. The language used by the participating agents,
   ii. The protocol followed by the agents as they negotiate, and
   iii. The decision process that each agent uses to determine its positions, concessions and criteria for agreement.
5. Any negotiation and bargaining mechanism should have the following attributes :
   i. **Efficiency :** The agents should not waste resources in coming to an agreement.
   ii. **Stability :** No agent should have an incentive to deviate from agreed-upon strategies.
   iii. **Simplicity :** The negotiation mechanism should impose low computational and bandwidth demands on the agents.
   iv. **Distribution :** The mechanism should not require a central decision maker.
   v. **Symmetry :** The mechanism should not be biased against any agent for arbitrary or inappropriate reasons.

PART-4

Argumentation Among Agents.

Questions-Answers

Long Answer Type and Medium Answer Type Questions

Que 4.10. Describe arguments in multiagent systems and its types.

Answer

Arguments in multi-agent systems :

1. Argumentation is defined as an activity aimed at convincing of the acceptability of a standpoint by putting forward propositions justifying or refuting the standpoint.
2. There are below factors involved in argumentation :
   i. Dialectic implies the structure of argumentation (that is acceptable arguments vs. fallacies).
   ii. Social psychology indicates agent attitude change and persuasion.
3. Artificial Intelligence formalize those approaches so as to provide formal theoretical results about particular models. This will automate agent or multi-agent argumentation capabilities.
4. For Arguments one needs to reason or provide justifications supporting a conclusion. It is represented as support → conclusion format.
5. Following are various types of arguments :
   a. Informational arguments : (Beliefs → Belief format)
      e.g. If it is cloudy, it might rain.
   b. Motivational argument : (Beliefs, Desires → Desire format),
      e.g. If it is cloudy and you want to get out then you don't want to get wet,
   c. Practical argument : (Belief, sub goals → goal format)
      e.g. If it is cloudy and you own a raincoat then put the raincoat,
   d. Social arguments : (Social commitment → Goal, Desire format)
      e.g. I will stop at the corner because the law say so.
      e.g. I can't do that, I promise to my mother that I won't.
   e. Interactions (binary or collective) between arguments :
      1. Conflict (defeat) format
         e.g. attacks
         The conflict (defeat) format can be categories further as below :
         i. Rebut (symmetrical) :
            - support1 → condusion1 (e.g. Tweety is a bird -> tweety flies)
            - support2 → (not) conclusion1 (e.g. Tweety is a small bird -> tweety does not fly)
         ii. Undercut (asymmetrical) : defeat the assumptions or their link to the conclusion
            - support2 → (not) support1

e.g. no Tweety is not a bird, it is just a cartoon

2. There are interactions of support-type that are used for collective binary arguments in multi-agent system.

PART-5

Trust and Reputation in Multiagent Systems.

Questions-Answers

Long Answer Type and Medium Answer Type Questions

Que 4.11. What do you understand by trust and reputation in multiagent systems ?

Answer

Trust and reputation in multi-agent systems and their societies :

Belief maintenance in multi-agent system :

1. In a multi-agent system, a truth maintenance system (TMS) is designed to ensure the integrity of an agent's knowledge, which should be stable, well founded and logically consistent.
2. Depending on how beliefs, justifications, and data are represented, a stable state of a knowledge base is defined in which :
   i. each datum that has a valid justification is believed, and
   ii. each datum that lacks a valid justification is disbelieved.
3. A well-founded knowledge base permits no set of its beliefs to be mutually dependent.
4. A logically consistent knowledge base is one that is stable at the time that consistency is determined and in which no logical contradiction exists.
5. A consistent knowledge base is one in which no datum is both believed and disbelieved or in which no datum and its negation are both believed.
6. A knowledge base should be always complete, concise, accurate, and efficient.

Societies of agents in multi-agent system :

1. In a multi-agent system, a group of agents can form a small society in which they play different kinds of roles. The group defines the roles, and the roles define the commitments associated with them.

2. When an agent joins a group, he joins in one or more roles, and acquires the commitments of that role.
3. Social commitments are the commitments of an agent to another agent that in turn define social dependence.
4. Cooperation is a form of such mutual dependence.
5. Social exchange is a form of such reciprocal dependence.

5 UNIT

Applications

CONTENTS

PART-1

AI applications, Language Models.

Questions-Answers

Long Answer Type and Medium Answer Type Questions

Que 5.1. Describe the applications of artificial intelligence.

Answer

Applications of artificial intelligence :

1. **Gaming :** AI plays crucial role in strategic games such as chess, poker, tic-tac-toe, etc., where machine can think of large number of possible positions based on heuristic knowledge.
2. **Natural language processing :** It is possible to interact with the computer that understands natural language spoken by humans.
3. **Expert systems :** There are some applications which integrate machine, software, and special information to impart reasoning and advising. They provide explanation and advice to the users.
4. **Vision systems :** These systems understand, interpret, and comprehend visual input on the computer.
5. **Speech recognition :** Some intelligent systems are capable of hearing and comprehending the language in terms of sentences and their meanings while a human talks to it. It can handle different accents, slang words, noise in the background, change in human's noise due to cold, etc.
6. **Handwriting recognition :** The handwriting recognition software reads the text written on paper by a pen or on screen by a stylus. It can recognize the shapes of the letters and convert it into editable text.
7. **Intelligent robots :** Robots are able to perform the tasks given by a human. They have sensors to detect physical data from the real world. They have efficient processors, multiple sensors and huge memory, to exhibit intelligence.

Que 5.2. What are various types of language models ?

Answer

Various kind of knowledge can be captured through the use of a small number of formal models or theories. Models and theories are all drawn for the standard tool-kit of computer science, mathematics and linguistics.

Types of Models :

1. **State machines :** It is based on simplest formulation of state, transition among states, input representation. Examples of state machines are finite-state automata (Deterministic, Non deterministic), finite-state transducers.
2. **Rule systems :**
   i. Rule system can be define using regular grammars, context-free grammars, feature augmented grammars.
   ii. State machines and formal rule systems are the main tools used when dealing with knowledge of phonology, morphology and syntax.
3. **Logic :**
   i. The logical representations have traditionally been used for modeling semantics and pragmatics, although more recent work has tended to focus on potentially more robust techniques drawn from non-logical lexical semantics.
   ii. First order logic / predicate calculus, lambda-calculus, feature structures, semantic primitives are used for logical representation.
4. **Probabilistic models :**
   i. These are crucial for capturing every kind of linguistic knowledge. Each of the other models can be augmented with probabilities.
   ii. The state machine augmented with probabilities can become :
      a. Weighted automaton, or Markov model.
      b. Hidden Markov Models (HMMs).
5. Vector-space models are based on linear algebra and information-retrieval. It has good use in Word meanings.

PART-2

Information Retrieval, Information Extraction.

Questions-Answers

Long Answer Type and Medium Answer Type Questions

Que 5.3. Explain information retrieval with its characteristics.

Answer

Information retrieval is concept associated with subjects like database management system, artificial intelligence, because, in both stream we have

huge collection of data (DBMS with database or data warehouses, AI with knowledge base).

Definition :

Information retrieval is process of sending documents / text / data in which the user is interested.

For example, information retrieval is widely used in popular search engine like Google, Yahoo, Bing etc.

Characteristics of Information Retrieval :

1. **A huge data/document collection :** In many systems, we need to decide boundaries such as document, paragraph or a page or a multi-page text, from huge data collection.
2. **A format of query with standard query language :** It is imposed by user but if standard language is used then user need to follow syntax.
3. **The generated result model :** Appropriate text/data/document are generated by the information retrieval system with specific query.
4. **Displaying results model :** The display of the result can be very simple as that of title of document or highly graphical and complex representation (i.e., 3-D information).

Que 5.4. What is information extraction ?

Answer

Information extraction :

1. Information extraction is the technique of creating database entries.
2. The database entries are done by overlooking the text and its occurrences for specific class of object or event and for relationship among those objects and events.
3. The field such as street, city, state, pin-code are extracted from instances of addresses from web pages.
4. The instance of weather report with temperature, wind, speed, humidity are extracted.
5. Information retrieval systems and full text parser can be mapped by mid system called information extraction.

Que 5.5. Discuss the component of a typical relational based extraction-FASTUS.

Answer

The component of typical system FASTUS are :

1. **Tokenization :**
   i. It is first stage of FASTUs which converts the stream of character into tokens (words, numbers and punctuation).

ii. The white space, a punctuation separate the tokens.

iii. It is easy to process English. But in Japanese language, we need to perform segmentation using Viterbi segmentation algorithm.

iv. Using the markup language such as HTML and XML tokenization process is performed.

2. **Complex words handling :**

i. The difficult complex words are processed using "set up" joint venture and name system such as "Prime Minister Narendra Modi" "President Ram Nath Kovind" and "Quantum Page Pvt. Ltd".

ii. The naming system can be developed using finite-state grammar rules.

3. **Basic group handling :**

i. The grouping of form head such as noun group and verb group handling is called basic group handling.

ii. The grouping into units can be utilized in further stages. Noun phrase is having head noun.

4. **Complex phrase handling :**

i. The grouping of basic group into complex phrases is handled in complex phase.

ii. The aim of this component is to design finite-state rule.

5. **Structure merging :** It is a phase which merges the complex phrases.

PART-3

Natural Language Processing.

Questions-Answers

Long Answer Type and Medium Answer Type Questions

Que 5.6. What do you mean by natural language processing ? Why it is needed ?

Answer

1. Natural Language Processing (NLP) studies the problems inherent in the processing and manipulation of natural language and to make computer understand statements written in human language.

2. NLP can be defined as the automatic processing of human language.

3. Natural language processing is a subfield of AI which deals with the methods of communicating with a computer in one's own natural language.
4. It is used for analyzing and representing naturally occurring texts at one or more levels of linguistic analysis for the purpose of achieving human-like language processing for a range of tasks or applications.
5. It is needed to bridge the gap between human and machine.
6. The goal of natural language is to enable people and computers to communicate in a natural language, such as English.
7. The field of NLP is divided into subfields :
   a. **NLU (Natural Language Understanding) :** This investigates methods of allowing the computer to comprehend instructions given in English.
   b. **NLG (Natural Language Generation) :** This strive that computer produce ordinary English language so that people can understand computers more easily.
8. The study of language generation falls into following three areas :
   a. Determination of content.
   b. Formulating and developing a text utterance plan.
   c. Achieving a realization of the desired utterances.
9. A full NLU system would be able to :
   a. Paraphrase an input text.
   b. Translate the text into another language.
   c. Answer questions about the contents of the text.
   d. Draw inferences from the text.
10. Applications of NLP :
    a. Natural language interfaces to databases.
    b. Machine translation.
    c. Advanced word-processing tools.
    d. Explanation generation for expert systems.

Que 5.7. What are the applications of natural language processing ?

Answer

Following are the applications of natural language processing :

i. Machine Translation :

1. As the amount of information available online is growing, the need to access it becomes increasingly important.

2. Due to this the value of natural language processing applications becomes clear.
3. Machine translation helps us in translating technical manuals, support content or catalogs at a significantly reduced cost.
4. The challenge with machine translation technologies is not in translating words, but in understanding the meaning of sentences to provide a true translation.

ii. Automatic summarization :

1. When we access a specific, important piece of information from a huge knowledge base; information overload can become a problem.
2. Automatic summarization is relevant for summarizing the meaning of documents and information.
3. Automatic summarization is helpful in understanding the emotional meanings inside the information, such as in collecting data from social media.
4. Automatic summarization is especially relevant when used to provide an overview of a news item or blog posts, while avoiding redundancy from multiple sources and maximizing the diversity of content obtained.

iii. Sentiment analysis :

1. The goal of sentiment analysis is to identify sentiment among several posts where emotion is not always explicitly expressed.
2. Companies use sentiment analysis, to identify opinions and sentiment online to help them understand what customers think about their products and services.
3. Sentiment analysis understands sentiments in context to help you better understand what's behind an expressed opinion, which can be extremely relevant in understanding and driving purchasing decisions.

iv. Text classification :

1. Text classification makes it possible to assign predefined categories to a document and organize it to help you find the information you need.
2. For example, an application of text categorization is spam filtering in email.

v. Question Answering :

1. Question-Answering (QA) is becoming popular due to applications such as Siri, OK Google, chat boxes and virtual assistants.

2. A QA application is a system capable of coherently answering a human request.
3. It may be used as a text-only interface or as a spoken dialog system.
4. While they offer great promise, they still have a long way to go.
5. Using natural language processing for creating a seamless and interactive interface between humans with machines will continue to be a top priority for various cognitive applications.

Que 5.8. What are the various steps in natural language processing ?

Answer

Steps in natural language processing :

1. **Morphological analysis :** It is concern with the way the words are built up from smaller, meaning bearing units; (come(s), co(mes)). Individual words are analyzed into their components and non-word tokens such as punctuation are separated from the words.
2. **Syntactic analysis :** It is concern with how the words are put together to form correct sentences and what structural role each word has. Linear sequences of words are transformed into structures that show how the words relate to each other.
3. **Semantic analysis :** It is concern with what words mean and how these meanings combine in sentences to form sentence meanings. That is the structures created by the syntactic analyzer are assigned meanings.
4. **Pragmatic analysis :** It is concern with how sentences are used in different situations and how use affects the interpretation of the sentence. The structure representing what was said is reinterpreted to determine what was actually meant.
5. **Discourse integration :** It is concern with how immediately preceding sentences affect the interpretation of the next sentence. The meaning of an individual sentence may depend on the sentences that precede it and may influence the meanings of the sentences that follow it.

Que 5.9. Describe the role of artificial intelligence in natural language processing. **AKTU 2017-18, Marks 10**

Answer

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

2. AI provides computer the ability to accept spoken words as dictation or to follow voice commands by using software.
3. AI programs are able to communicate with humans in a natural way because natural language is one of the most important medium for the communication.
4. To understand the natural language, a program needs a considerable knowledge about the structure of the language including what the words are and how they combine into phrases and sentences.
5. There are three different approaches for the development of natural language understanding programs :
   i. The use of keyword and pattern matching.
   ii. The use of combined syntactic (structural) and semantic directed analysis.
   iii. Comparing and matching the input to real world situation.

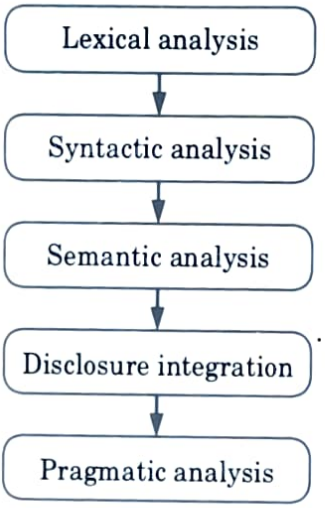
Que 5.10. What is the role of NLP in AI ? Illustrate the various phases in NLP. **AKTU 2018-19, Marks 10**

Answer

Role of NLP : Refer Q. 5.9, Page 5–9A, Unit-5.

There are five phases involved in NLP :

1. **Lexical analysis :** It involves identifying and analyzing the structure of words. Lexical analysis is dividing the whole chunk of text into paragraphs, sentences, and words.
2. **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.
3. **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.
4. **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 the sentence that follows it.
5. **Pragmatic analysis :** During pragmatic analysis, what was said is re-interpreted on what it actually meant. It involves deriving those aspects of language which require real world knowledge.



PART-4

Machine Translation.

Questions-Answers

Long Answer Type and Medium Answer Type Questions

Que 5.11. What do you understand by machine translation ? What are various types of machine translation systems ?

Answer

Machine translation :

1. Machine translation is the task of automatically converting source text in one language to text in another language.
2. In a machine translation task, the input already consists of a sequence of symbols in some language, and the computer program must convert this into a sequence of symbols in another language.

Types of Machine Translation Systems :

1. **Bilingual MT System :** Bilingual MT systems produce translations between two particular languages.
2. **Multilingual MT System :** Multilingual MT systems produce translations between any pair of languages. They may be either uni-directional or bi-directional in nature.

Que 5.12. **What are the three major approaches of machine translation ?**

Answer

Three major approaches are :

1. **Direct MT Approach :** It is less popular but the oldest approach of MT. The systems that use this approach are capable of translating SL (source language) directly to TL (target language). Such systems are bi-lingual and uni-directional in nature.

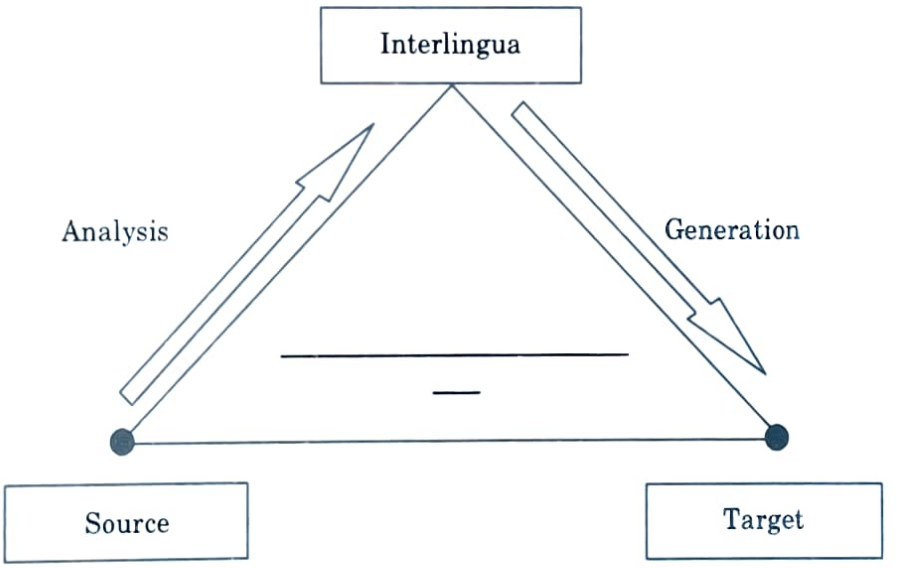


Fig. 3.12.1.

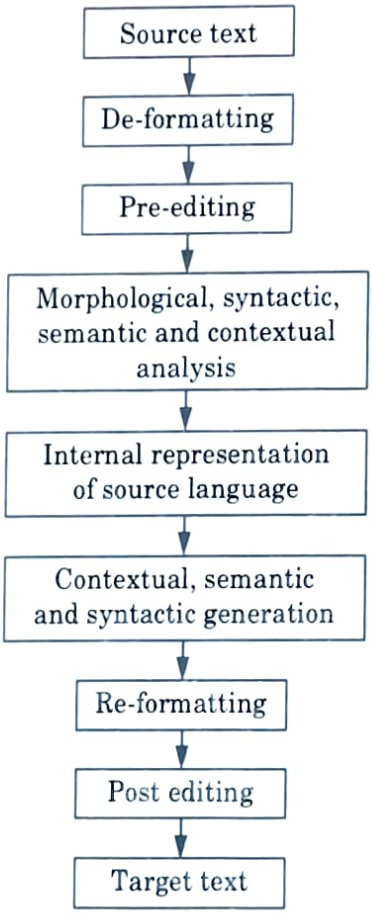
2. **Interlingua Approach :** The systems that use Interlingua approach translate SL to an intermediate language called Interlingua (IL) and then translate IL to TL. The Interlingua approach can be understood with the help of the following MT pyramid.
3. **Transfer Approach :** Three stages are involved with this approach :
   a. In the first stage, source language (SL) texts are converted to abstract SL-oriented representations.
   b. In the second stage, SL-oriented representations are converted into equivalent target language (TL)-oriented representations.
   c. In the third stage, the final text is generated.
4. **Empirical MT Approach :** This is an emerging approach for MT. Basically, it uses large amount of raw data in the form of parallel corpora. The raw data consists of the text and their translations. Analogy-based,

example-based, memory-based machine translation techniques use empirical MT approach.

Que 5.13. Design the process of machine translation with the help of flowchart.

Answer

Process of machine translation with the help of the following flowchart :



PART-5

Speech Recognition.

Questions-Answers

Long Answer Type and Medium Answer Type Questions

Que 5.14. Discuss the term speech recognition.

Answer

1. Speech recognition is the process that enables a computer to recognize and respond to spoken words and then convert them in a format that the machine understands. The machine may then convert it into another form of data depending on the end-goal.
2. Speech recognition is widely used in digital assistants, smart speakers, smart homes, and automation for a variety of services, products, and solutions.

Que 5.15. What are the most commonly used algorithms for speech recognition ?

Answer

1. **Natural language processing (NLP) :**
   i. NLP is not a specific algorithm used in speech recognition; it is the area of artificial intelligence which focuses on the interaction between humans and machines through speech and text.
   ii. Many mobile devices incorporate speech recognition into their systems to conduct voice search (example : Siri) or provide more accessibility around texting.
2. **Hidden Markov models (HMM) :**
   i. Hidden Markov Models build on the Markov chain model, which stipulates that the probability of a given state hinges on the current state, not its prior states.
   ii. While a Markov chain model is useful for observable events, such as text inputs, hidden Markov models allow us to incorporate hidden events, such as part-of-speech tags, into a probabilistic model.
   iii. They are utilized as sequence models within speech recognition, assigning labels to each unit i.e., words, syllables, sentences, etc., in the sequence.
   iv. These labels create a mapping with the provided input, allowing it to determine the most appropriate label sequence.
3. **Neural networks :**
   i. For deep learning algorithms, neural networks process training data by mimicking the interconnectivity of the human brain through layers of nodes.
   ii. Each node is made up of inputs, weights, a bias (or threshold) and an output.

iii. If that output value exceeds a given threshold, it "fires" or activates the node, passing data to the next layer in the network.

iv. Neural networks learn the mapping function through supervised learning, adjusting based on the loss function through the process of gradient descent.

v. While neural networks tend to be more accurate and can accept more data, this comes at a performance efficiency cost as they tend to be slower to train compared to traditional language models.

Que 5.16. What are the applications of speech recognition ?

Answer

1. **Automotive :** Speech recognizer improves driver safety by enabling voice-activated navigation systems and search capabilities in car radios.
2. **Technology :** Virtual assistants are increasingly becoming integrated within our daily lives, particularly on our mobile devices. We use voice commands to access them through our smartphones, such as through Google Assistant or Apple's Siri, for tasks, such as voice search, or through our speakers, via Amazon's Alexa or Microsoft's Cortana, to play music.
3. **Healthcare :** Doctors and nurses leverage dictation applications to capture and log patient diagnoses and treatment notes.
4. **Sales :** It can help a call center transcribe thousands of phone calls between customers and agents to identify common call patterns and issues. Cognitive bots can also talk to people via a webpage, answering common queries and solving basic requests without needing to wait for a contact center agent to be available. In both instances speech recognition systems help reduce time to resolution for consumer issues.
5. **Security :** As technology integrates into our daily lives, security protocols are an increasing priority. Voice-based authentication adds a viable level of security.

Que 5.17. What are the Techniques for Speech Recognition ?

Answer

Techniques for Speech Recognition are :

1. **Speech Analysis Technique :**
   i. Speaker identity can be shown by a different type of information that is present in speech data.
   ii. This incorporates speaker-specific information due to the vocal tract, excitation source, and behavior feature.
   iii. This stage deals with a suitable frame size for segmenting speech signals for further analysis and extracting.

2. Feature Extraction Technique :

i. The speech feature extraction technique is the process of placing words in groups or classes and decreasing the dimensionality of the input vector while maintaining the discriminating power of the signal.

ii. From the basic formation of speaker identification and verification system, we know that the number of training and test vector needed for the classification problem grows with the dimension of the given input; therefore, we need feature extraction of the speech signal.

3. Modeling :

i. The modeling technique aims to create speaker models using a speaker-specific feature vector. Further, Speaker recognition and Speaker identification are the parts of Modeling.

ii. The speaker identification technique identifies by itself, who is speaking based on individual information integrated into a speech signal.

PART-6

Robot.

Questions-Answers

Long Answer Type and Medium Answer Type Questions

Que 5.18. What are Robots ? What is Robotics ?

Answer

Robots :

i. Robots are the artificial agents acting in real world environment.

ii. 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.

Robotics :

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

ii. The robots have mechanical construction, form, or shape designed to accomplish a particular task.

iii. They have electrical components which power and control the machinery.

iv. They contain some level of computer program that determines what, when and how a robot does something.

Que 5.19. Difference between AI program and Robot System ?

Answer

S. No.	AI programs	Robots
1.	They usually operate in computer-stimulated worlds.	They operate in real physical world.
2.	The input to an AI program is in symbols and rules.	Input to robots is analog signal in the form of speech waveform or images.
3.	They need general purpose computers to operate on.	They need special hardware with sensors and effectors.

Que 5.20. How do Robots and Artificial Intelligence work together ?

Answer

i. Artificial Intelligence gives robots a computer vision to navigate, sense and calculate their reaction accordingly.

ii. Robots learn to perform their tasks from humans through machine learning which again is a part of computer programming and AI.

iii. AI has the power to give life to robots and empower them to take their decisions on their own.

iv. Depending on the use and the tasks that the robot has to perform different types of AI is used. They are as follows :

1. **Weak Artificial Intelligence :**

   i. This type of AI is used to create a simulation of human thought and interaction. The robots have predefined commands and responses.

   ii. The robots do not understand the commands they do only the work of retrieving the appropriate response when the suitable command is given.

   iii. The example of this is Siri and Alexa. The AI in these devices only executes the tasks as demanded by the owner.

2. **Strong Artificial Intelligence :**
   i. This type of AI is used in those robots that perform their tasks on their own. They do not need any kind of supervision once they are programmed to do the task correctly.
   ii. This type of AI is widely used in automated things and the examples are self-driving cars and internet cars.
   iii. This type of AI is also used in humanoid robots which can sense their environment quite well and interact with their surroundings.
   iv. Also, robotic surgeons are becoming popular day by day as there is no human intervention required at all.
3. **Specialized Artificial Intelligence :**
   i. This type of AI is used when the robot needs to perform only specified special tasks. It is restricted only to limited tasks.
   ii. This includes mainly **industrial robots** which perform specified and repetitive tasks like painting, tightening, etc.

Que 5.21. What are the Components of a Robot ?

Answer

Robots are constructed with the following :

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

Que 5.22. Describe robot locomotion. What are types of locomotion ?

Answer

Locomotion is the mechanism that makes a robot capable of moving in its environment. There are various types of locomotion :

1. **Legged Locomotion :**
   i. This type of locomotion consumes more power while demonstrating walk, jump, trot, hop, climb up or down, etc.
   ii. 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.
   iii. 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.
2. **Wheeled Locomotion :**
   i. It requires fewer numbers of motors to accomplish a movement.
   ii. 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.
      a. **Standard wheel :** Rotates around the wheel axle and around the contact.
      b. **Castor wheel :** Rotates around the wheel axle and the offset steering joint.
      c. **Swedish 45° and Swedish 90° wheels :** Omni-wheel, rotates around the contact point, around the wheel axle, and around the rollers.
      d. **Ball or spherical wheel :** Omnidirectional wheel, technically difficult to implement.
3. **Slip/Skid Locomotion :**
   i. In this type, the vehicles use tracks as in a tank.
   ii. 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.

PART-7

Hardware, Perception, Planning, Moving.

Questions-Answers

Long Answer Type and Medium Answer Type Questions

Que 5.23. **Describe mobile robot hardware.**

Answer

A mobile robot is nothing but combination of various hardware and computational components. In terms of hardware components, it is a collection of subsystems like.

1. **Motion :** How the robot changes his position w.r.t an object over time in its environment.
2. **Sensing :** How the robot measures properties of itself as well as detects its environment.
3. **Reasoning :** How the robot maps these measurements into actions.
4. **Communication :** How the robot communicates information with an outside operator.

Que 5.24. Explain the categories of motion of mobile robot.

Answer

Following are the four categories of motion of mobile robot :

1. **Terrestrial :**
   i. Terrestrial robots move on the ground.
   ii. They are structured to take benefit of a solid support surface as well as gravity.
   iii. Wheeled robots are most common type of robots in this category. They can walk, climb, roll, use tracks, or slither to move. Terrestrial robots are also known as ground-contact robots.
2. **Airborne :**
   i. Robotic helicopters, fixed-wing aircraft, robotically controlled parachutes, and dirigibles have been developed.
   ii. Airborne robot vehicles often mimic existing aircraft or birds.
   iii. Flying robots share many issues with aquatic robots.
3. **Aquatic :**
   i. This type of robots operates in water, either at the surface or underwater.
   ii. Such types of aquatic vehicles are not only used in water jets but also in propellers to provide locomotion.
   iii. Potentially important application domain since not only is most of Earth's surface covered with water; but much of the ocean is not readily accessible to humans.
4. **Space :**
   i. There are two main classes of robot those that move by climbing (over a larger vehicle) as well as those that are independently propelled (known as free flyers).

ii. Some robots are designed to operate in the microgravity of outer space, typically for space station maintenance.

iii. Various locomotive devices enable these robots to move about their environment.