SNS COLLEGE OF ENGINEERING

Kurumbapalayam(Po), Coimbatore – 641 107
Accredited by NAAC-UGC with 'A' Grade
Approved by AICTE, Recognized by UGC & Affiliated to Anna University, Chennai

Department of Artificial Intelligence and Data Science

Course Name – Introduction to Artificial Intelligence

II Year / III Semester

Unit 1 Problems and Techniques,AI Programming Language

10-Aug-22

Problem Solving Techniques

Artificial Intelligence is beneficial for solving complex problems due to its efficient methods of solving. Following are some of the standard problem-solving techniques used in AI. You can explore about other problem-solving techniques apart from searching.

Heuristics

The heuristic method helps comprehend a problem and devises a solution based purely on experiments and trial and error methods. However, these heuristics do not often provide the best optimal solution to a specific problem. Instead, these undoubtedly offer efficient solutions to attain immediate goals. Therefore, the developers utilize these when classic methods do not provide an efficient solution for the problem. Since heuristics only provide time-efficient solutions and compromise accuracy, these are combined with optimization algorithms to improve efficiency.

Example: Travelling Salesman Problem

The most common example of using heuristic is the Travelling Salesman problem. There is a provided list of cities and their distances. The user has to find the optimal route for the Salesman to return to the starting city after visiting every city on the list. The greedy algorithms solve this NP-Hard problem by finding the optimal solution. According to this heuristic, picking the best next step in every current city provides the best solution.

Searching Algorithms

Searching is one of the primary methods of solving any problem in AI. Rational agents or problem-solving agents use these searching algorithms to find optimal solutions. These problem-solving agents are often goal-based and utilize atomic representation. Moreover, these searching algorithms possess completeness, optimality, time complexity, and space complexity properties based on the quality of the solution provided by them.

Types of Searching Algorithms

There are following two main types of searching algorithms:

1. Informed Search
2. Uninformed Search

Informed Search

These algorithms use basic domain knowledge and comprehend available information regarding a specified problem as a guideline for optimal solutions. The solutions provided by informed search algorithms are more efficient than uninformed search algorithms.

Types of informed Search Algorithms

There are following main two types of informed search algorithms:

1. Greedy Search
2. A * Search

Uninformed Search

These algorithms do not have the privilege of using basic domain knowledge, such as the desired goal's closeness. It contains information regarding traversing a tree and identifying leaf and goal nodes. Uninformed search also goes by the name of blind search because while traversing, there is no specific information about the initial state and test for the goal. This search goes through every node till reaching the desired destination.

Types of Uninformed Searching Algorithms

There are the following main five types of uninformed search algorithms:

1.Breadth-First Search
2.Depth First Search
3.Uniform Cost Search
4.Iterative Deepening Depth First Search
5.Bidirectional Search

Why Problem Solving is Important in AI?

Decoding any type of problem needs specific organized measures to be observed. Identical is the matter of solving issues by AI. The following are the details:

- **Goal** –In this phase as soon as a crisis appears, the AI agent puts a goal or a mark. This needs the agent to thoroughly examine and clarify the issue. This is a vital action as if the goal for the issue is poorly developed then all the actions carried out to achieve the goal would be useless.
- **Problem Description**–This is one major stage of problem-solving. Whenever a problem occurs, then the agent chooses what measures must be carried out to run to the developed goal. This is accomplished in the subsequent actions:
- **Describing the State**–A state area can be described as a group of all the accurate conditions in which an agent can be joined when discovering a key to the crisis.
  - **Specifying Primary State**–For an agent to begin cracking the issue, it must begin from a state. The primary state from where the agent begins performing is directed to the primary state.
  - **Collect Details**–Now the agent collects data and utilizes the data needed by it to fix the issue. These details will be collected with one-time incidents as well as present pieces of knowledge.
  - **Designing the Changes**–Some issues are undersized and so these can be deciphered efficiently. But most of the time issues will be such where sound planning and implementation are required. Hence this needs appropriate data structures and managing processes well in advance.

The lines that are seen between these layers are called ‘weights’. Each one of them usually represents a float number, or a decimal number, which is multiplied by the value in the input layer. All the weights add up in the hidden layer. The dots in the hidden layer represent a value based on the sum of the weights. These values are then passed to the next hidden layer.

Output Layer:

The output layer gives us segregated photos. Once the layer adds up all these weights being fed in, it'll determine if the picture is a portrait or a landscape.

Example - Predicting Airfare Costs

This prediction is based on various factors, including:

•Airline

•Origin airport

•Destination airport

•Departure date

We begin with some historical data on ticket prices to train the machine. Once our machine is trained, we share new data that will predict the costs. Earlier, when we learned about four kinds of machines, we discussed machines with memory. Here, we talk about the memory only, and how it understands a pattern in the data and uses it to make predictions for the new prices as shown.

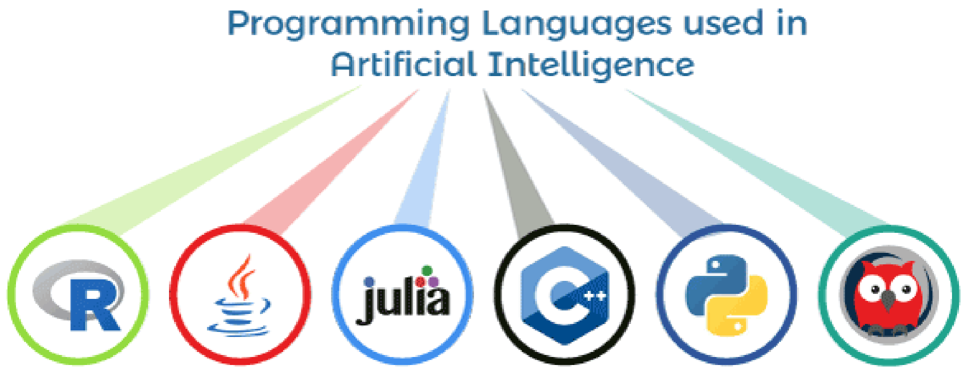
Steps of problem solving in AI

Performance benchmark is one of the vital things in AI problem solving which determines the value of the algorithm utilized to fix the issue. There are four methods in which the execution of an algorithm is calculated. These are as follows:

- **Totality** – Totality calculates the algorithm’s assurance to discover the solution for the issue if there is any solution for it.
- **Optimality** – This step is utilized to calculate the tracking methods which discover an optimal solution to the presented issue.
- **Duration Complexity** – This calculates the portion of time the algorithm abides to run till the key for the assigned issue.
- **Space Intricacy** – This is the standard that is employed to specify the quantity of area (in memory) that the algorithm needs to execute the quest.

Languages used in Artificial Intelligence

Artificial Intelligence has become an important part of human life as we are now highly dependent on machines. **Artificial Intelligence is a very important technology to develop and build new computer programs and systems, which can be used to simulate various intelligence processes like learning, reasoning, etc.**


Programming Languages used in Artificial Intelligence
R
julia
C++