Smart systems

Eng. Khadija M.Hesham

AI Definitions

The branch of computer science that is concerned with the automation of intelligent behavior

AI as a field of study of

- Computer Science
- Cognitive Science
- Psychology
- Philosophy
- Linguistics
- Neuroscience

Intelligence

AI is a belief that the brain is a form of biological computer and that the mind is computational

Definition of intelligence:

the ability to comprehend; to understand and profit from experience

The Turing Test

1950 – Alan Turing devised a test for intelligence called the Imitation Game

Ask questions of two entities, receive answers from both

If you can’t tell which of the entities is human and which is a computer program, then you are fooled and we should therefore consider the computer to be intelligent

Solving the Turing Test

Created by Joseph Weizenbaum

Weizenbaum wrote the program so that it would generate an English response/question based on a group of patterns

If the user sentence matched a pattern, this pattern would be used to generate the next sentence/question

Table-Lookup vs. Reasoning

Consider two approaches to programming a Tic-Tac-Toe player

Solution 1: a pre-enumerated list of best moves given the board configuration

Solution 2: rules (or a heuristic function) that evaluate a board configuration, and using these to select the next best move

Brain vs. Computer

In AI, we compare the brain (or the mind) and the computer

Our hope: the brain is a form of computer

Our goal: we can create computer intelligence through programming just as people become intelligent by learning

Intelligence Problems

The computer is that it works strictly syntactically.

Ex:

An opcode is the first byte of an instruction in machine language which tells the hardware what operation needs to be performed with this instruction.

Op code: 10011101 translates into a set of microcode instructions such as: signal memory read.

There is no understanding

x = y + z;   is meaningless to the computer

the computer doesn’t understand addition, it just knows that a certain op code means to move values to the adder and move the result elsewhere

AI Assumptions

- We can understand and model cognition without understanding the underlying mechanism.

  That is, it is the model of cognition that is important not the physical mechanism that implements it

  If this is true, then we should be able to create cognition (mind) out of a computer or a brain or even other entities that can compute such as a mechanical device

  This is the assumption made by symbolic AI researchers

AI Assumptions

- Cognition will emerge from the proper mechanism

That is, the right device, fed with the right inputs, can learn and perform the problem solving that we, as observers, call intelligence

This is the assumption made by connectionist AI researchers

Symbolic AI Problems

1. **Scalability**

It can take dozens or more man-years to create a useful systems

It is often the case that systems perform well up to a certain threshold of knowledge (approx. 10,000 rules), after which performance (accuracy and efficiency) degrade.

Symbolic AI Problems

2. Brittleness

Solving a specific problem,

1. move away from that domain area and the system’s accuracy drops rapidly (no generalization).
2. No or little capacity to learn, so performance (accuracy) is static
3. **Lack of real-time performance**

Connectionist AI Problems

1. **No “memory” or sense of temporality**

The first problem can be solved to some extent (cloud storage)

2. **Learning is problematic**

Learning times can greatly vary

Overtraining leads to a system that only performs well on the training set and undertraining leads to a system that has not generalized (overfitting)

Connectionist AI Problems

3. No explicit knowledge-base

So there is no way to tell what a system truly knows or how it knows something (blackbox)

4. No capacity to explain its output

Explanation is often useful in an AI system so that the user can trust the system’s answer

Solutions

Most AI research has fallen into one of two categories

Select a specific problem to solve

1. study the problem (perhaps how humans solve it)
2. come up with the proper representation for any knowledge needed to solve the problem
3. acquire and codify that knowledge
4. build a problem solving system

Solutions

Select a category of problem or cognitive activity (e.g., learning, natural language understanding)

1. theorize a way to solve the given problem
2. build systems based on the model behind your theory as experiments
3. modify as needed

Solutions Requirements

- one or more representational forms for the knowledge
- some way to select proper knowledge, that is, search

Search Methodology

Search techniques are universal problem-solving methods.

Rational agents or Problem-solving agents in AI mostly used these search strategies or algorithms to solve a specific problem and provide the best result.

Problem-solving agents are the goal-based agents and use atomic representation.

Search Algorithm Terminologies

1. **Search:** Searching is a step by step procedure to solve a search-problem in a given search space. A search problem can have three main factors:

- **Search Space:** Search space represents a set of possible solutions, which a system may have.
- **Start State:** It is a state from where agent begins the search.
- **Goal test:** It is a function which observe the current state and returns whether the goal state is achieved or not.

Search Algorithm Terminologies

- **Search tree:** A tree representation of search problem is called Search tree. The root of the search tree is the root node which is corresponding to the initial state.
- **Actions:** It gives the description of all the available actions to the agent.
- **Transition model:** A description of what each action do, can be represented as a transition model.
- **Path Cost:** It is a function which assigns a numeric cost to each path.
- **Solution:** It is an action sequence which leads from the start node to the goal node.
- **Optimal Solution:** If a solution has the lowest cost among all solutions.

Properties of Search Algorithms:

Completeness: A search algorithm is said to be complete if it guarantees to return a solution if at least any solution exists for any random input.

Optimality: If a solution found for an algorithm is guaranteed to be the best solution (lowest path cost) among all other solutions, then such a solution for is said to be an optimal solution.

Time Complexity: Time complexity is a measure of time for an algorithm to complete its task.

Space Complexity: It is the maximum storage space required at any point during the search, as the complexity of the problem.

Types of search algorithms

Uninformed/Blind Search:

(also called a Heuristic search)

- It examines each node of the tree until it achieves the goal node(blind).
- does not contain any domain knowledge such as closeness, the location of the goal.
- It operates in a brute-force way as it only includes information about how to traverse the tree and how to identify leaf and goal nodes.
- search tree is searched without any information about the search space like initial state operators and test for the goal

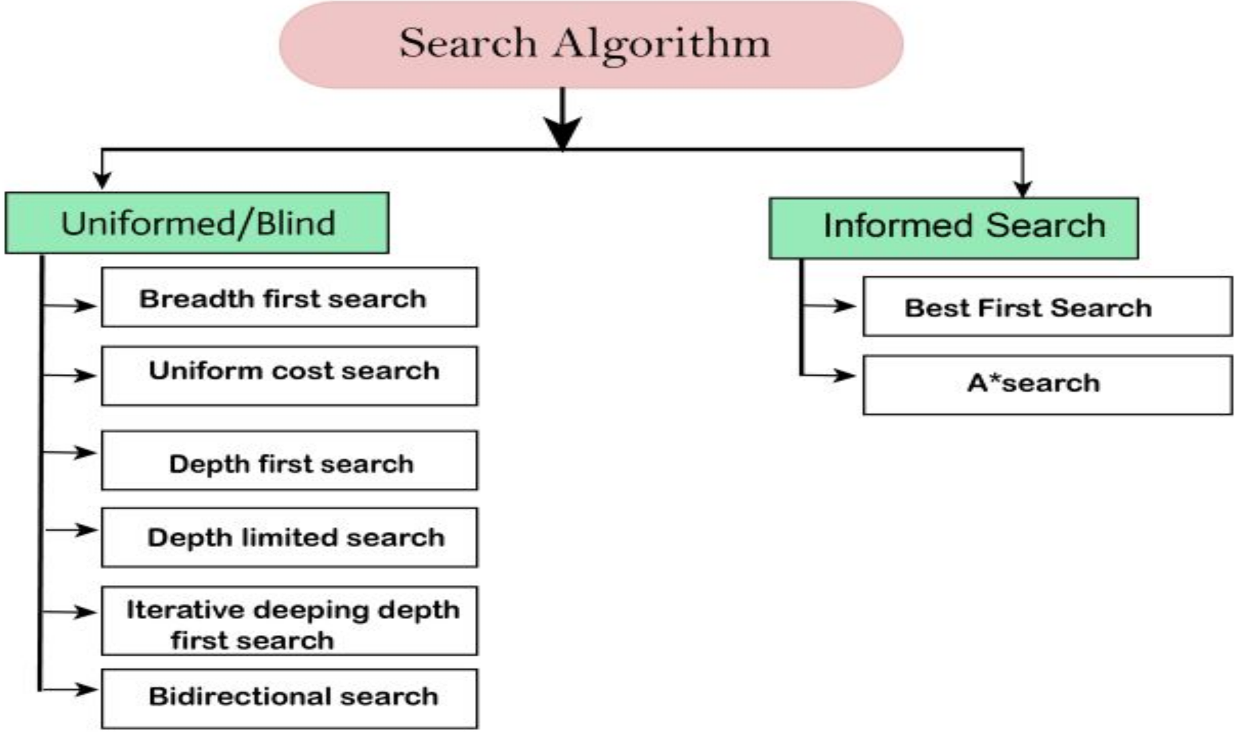
Types of search algorithms

Informed Search:

(also called a Heuristic search)

- Use domain knowledge.
- problem information is available.
- find a solution more efficiently than an uninformed search strategy.

Types of search algorithms



References

[1] Artificial Intelligence Introduction - nku.edu, CSC625, NOTES.

[2] https://www.javatpoint.com/search-algorithms-in-ai