AI

7/03/2025

* Definition of AI * Areas of use
↳ LLMs * Research areas in AI * Lack of system Bias * Advantages & Disadvantages of AI * ML.
- Relationship b/w ML & Dataset * features in programming
- Labels in Dataset * Types of Learning in ML

* A model is gotten when an Algorithm is trained using the datasets. Algorithms are trained while Models can be fine tuned

⇒ Types of Learning: (i) Supervised & Unsupervised learning algorithms: List them. (iii) Semi-Supervised (iv) Reinforcement learning

* Clustering & Data mining used in Unsupervised learning.

⇒ Data mining - used to identify & cluster datasets.

Building an AI system

(1) Data (2) Application scenario (3) ~~Data~~ Computing Power (4) Algorithm

⇒ Deep Learning * Problems of Datasets. * Properties of a good Dataset. * Deep learning algorithms.

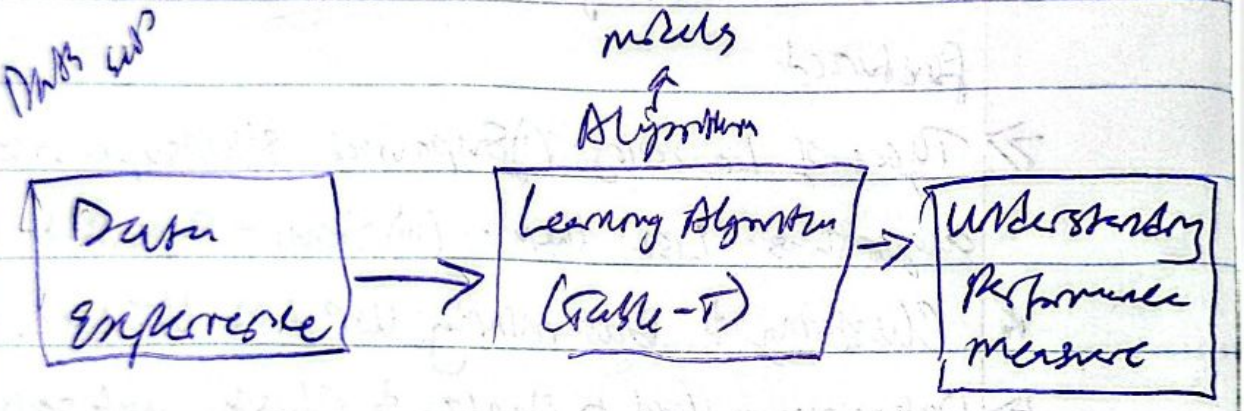
⇒ Generative AI - LLM's. * Advantages & Disadvantages

AI

28/03/2025

* Advantages & Disadvantages of AI
* Major areas of AI
* Generative AI

Machine Learning



- Machine learning is often confused with

- A computer program is set to learn from experience E in respect to ~~performance measure~~ P some class of task T & performance measure P if these performance of the task increases in experience

* Actual result & expected result
  ↳ System generate ↳ System result
* Error is the difference b/w Actual & expected

Data is biased by Quality, Quantity, widespread

* Problems of AI systems ⇒ (1) Bias.

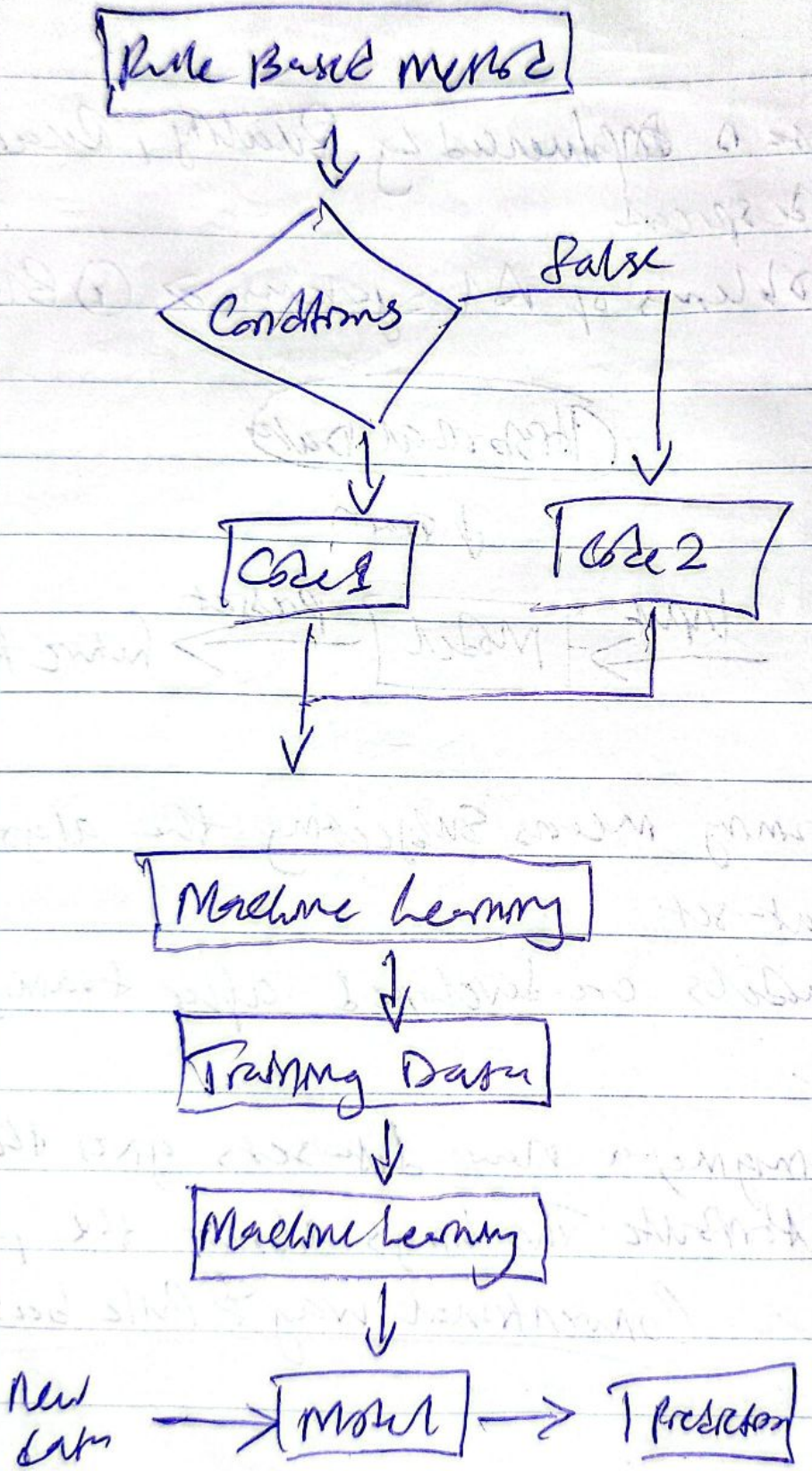
Historical Data

↓ train

New data → Input → Model → Predict → Future Attribute

- Training means subjecting the algorithms to dataset
- Models are developed after training the algorithms
- Bringing a new data sets gives the future attribute. This brings about the performance.

Conventional way & Rule based methods


Rule Based Model
Conditions
False
Code 1
Code 2
Machine learning
Training Data
Machine learning
New data
Model
Prediction

→ aABe

A → Abc | b

B → d

Sentence = abbcde

S → ABC → AbcBe → abbcBe → abbcde

→ abbcde → abbcBe → aAbcBe →

aABe → S

AI

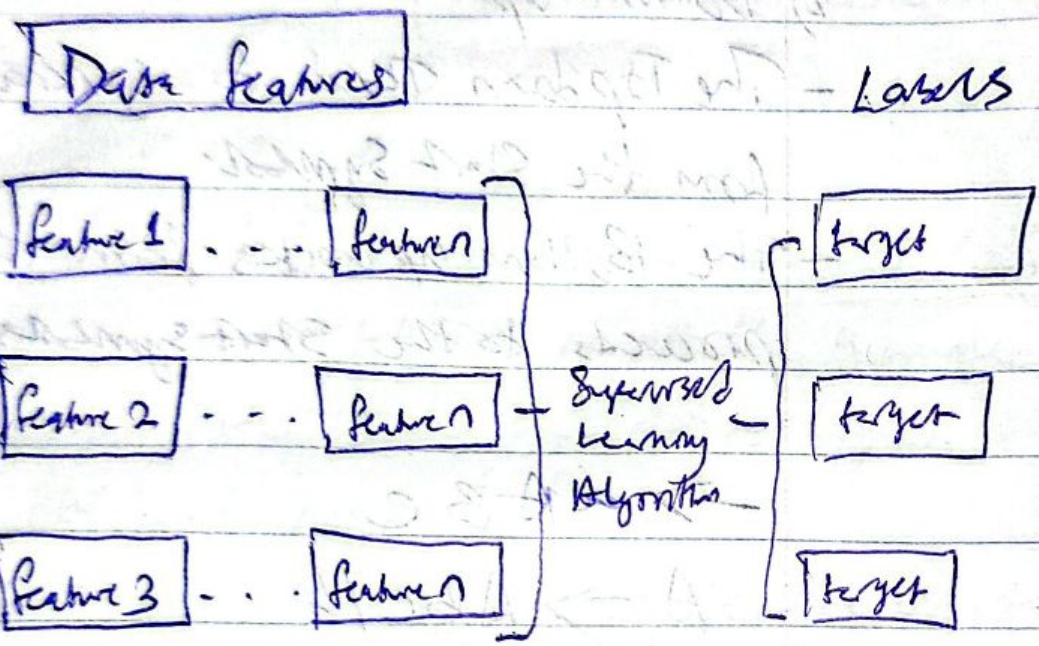
PROBLEMS SOLVED BY MACHINE LEARNING

1) Classification problems → 2 major problems
2) Regression Problems
3) Clustering problems

- Classification has discreet values
- Regression has Continuous values
- Clustering is

* Machine Learning → — Supervised
— Unsupervised
— Reinforcement Learning

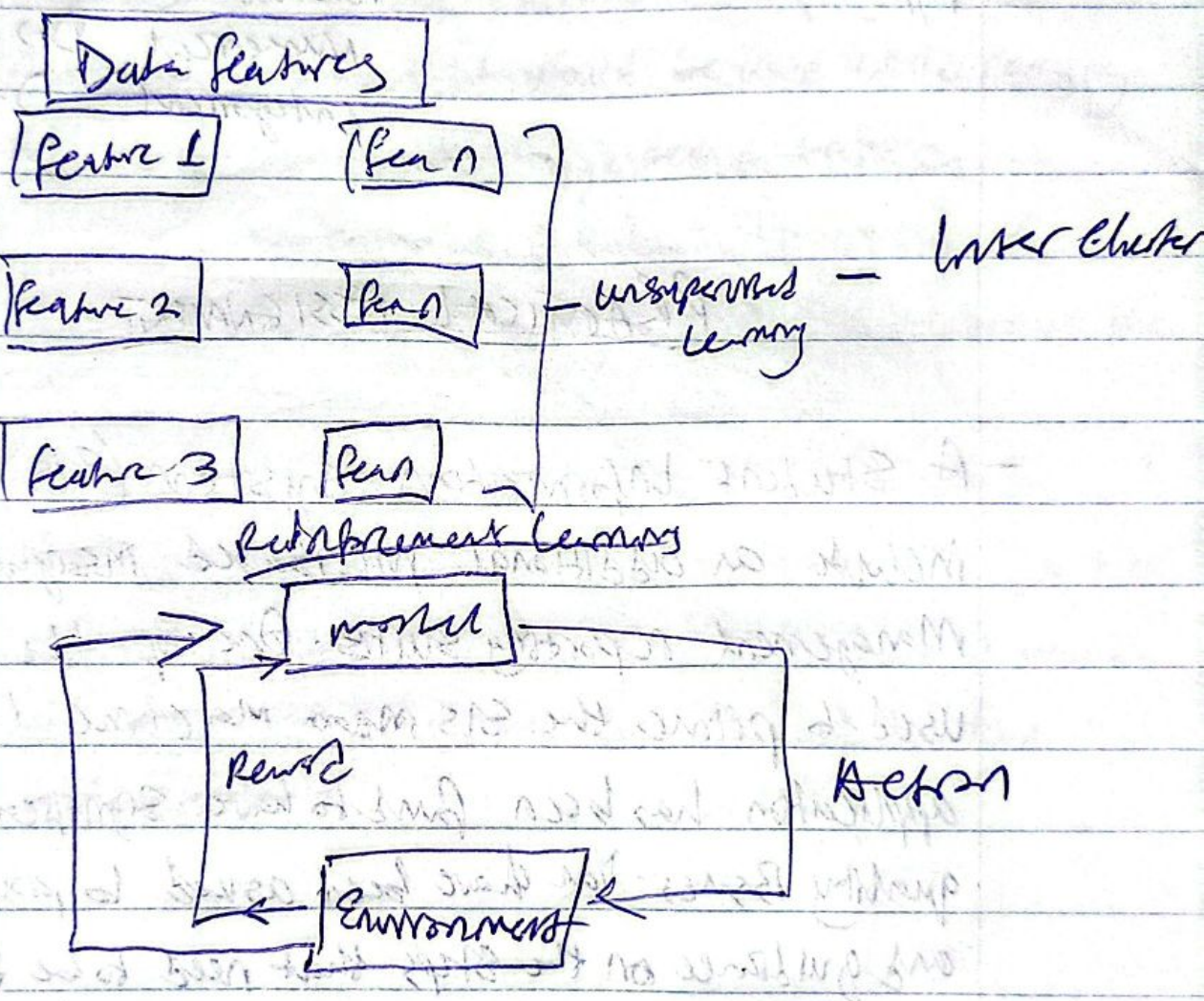
Supervised Learning



Weather	Temperature	Wind speed	Su
Sunny	High	High	
Rainy	low	medium	
Sunny	low	Low	

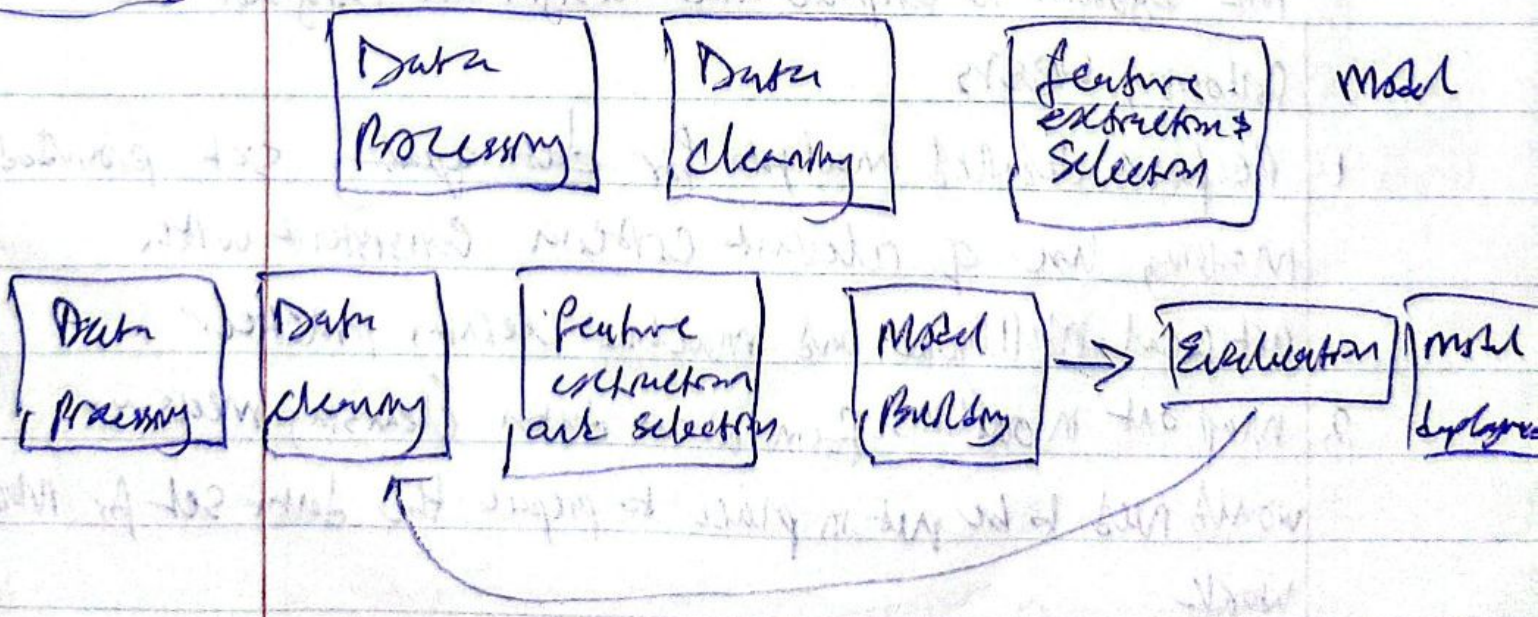
* classification problems are Yes or No responses

Unsupervised Learning Algorithms



Interactive exercise
Yes
No
Yes

Machine learning process



Literature Review

Types of Data set — Textual D-S
Numerical D-S
Categorical D-S

PRACTICAL ASSIGNMENT 3/04/2020

- A Student Information system (SIS) intends to include an additional ML Based insights into its Management reporting suite. One of the CSV data sets used to prime the SIS Machine Learning application has been found to have significant data quality issues. You have been asked to provide advice and guidance on the steps that need to be taken to clean the datafile as part of quality assurance for the ML System. To complete this assignment carry out the following tests

1 Perform detailed analysis for data quality set provided making use of relevant criteria consistent with artificial intelligence and Machine Learning practice

2 Map out in outline form what data cleansing measures would need to be put in place to prepare the data set for ML work

3 Indicate the types of PANDAS – SCIKIT_LEARN Python commands that would be used while carrying out the measures suggested in task 2

4 Attempt a cleanup of the Dataset using the procedures set out in task 2 & 3 and report on the degree of success in carrying this out.

- Submit the final document in MS Word containing my responses for sub task 1, 2, 3 & 4 in form of a report not less than 2500 words. The Python code must be included in the submission

ARTIFICIAL INTELLIGENCE

- AI is a branch of computer science that aims to create systems capable of performing tasks that typically require human intelligence. These tasks include reasoning, learning, perception, problem solving & decision making.

AI is categorized into 3 different categories

1) Narrow AI 2) weak AI 3) strong AI 4) Super intelligent AI

1) Narrow AI — Designed for specific tasks.
2) General AI — Is hypothetical AI with human like cognitive abilities
3) Super Intelligent AI —

By AI Techniques

1) Search and optimization : AI algorithm, Genetic algorithm, Knowledge Representation (Semantic network & ontologies), NLP's (Large language processing)
2) Computer vision: Object detection in self driving cars

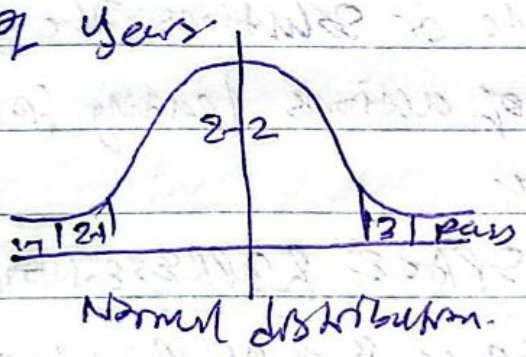
* ML is a subset of AI that enables systems to learn from data without explicit programming.

- ML algorithms improve performance through experience (learning of Datasets).

Types of ML Learning

- Supervised Learning. Eg
- Unsupervised Learning. Eg Anomalies detection.
- Reinforcement Learning - learns by interacting with an environment (rewards or penalties)

» NORMAL DISTRIBUTION - Established for thousands of years



Normal distribution.

Note Every event falls into a particular distribution.
eg Poisson distribution, Symmetrical

Eg of Reinforcement Learning; Autonomous Vehicles eg Tesla. Self driving cars use R-L to improve navigation Other examples are in education, healthcare.

- Note the basic the AI is used. i.e healthcare includes predictive, diagnosis, detection & prognosis. Forecasting.
- Ethical & Social Implications i.e Bias. This has exploded in facial Recognition system to pick specific

Challenges of AI

1) Bias & Racial discrimination
2) Jobs replacement, Automation
3) Privacy concerns, AI systems
- ~~Autocall~~

INTRODUCTION TO SEARCH TECHNIQUES

Search in AI Problem solving - Search is fundamental in AI for finding solutions in problems by exploring possible state or solutions. The goal is to find a sequence of actions leading from the initial state to the goal state

STATE SPACE REPRESENTATION

A state space is a graph where nodes represent states and edges represent transitions between states (Actions). The solution is found by navigating through this state

PROBLEM FORMULATION

i) Initial state
ii) Goal state
iii) Operators:

* Initial state — The state starting point of the problem
* Goal state — The desired outcome of the problem
* Operators — Is the actions that can transform from one state to another

Keys: Search methods;

1) Uninformed Search methods – eg-Breadth-first-search
   -depth-first-search
   -uniform cost search
2) Informed Search methods

Implementation

```
from collections import deque
def bfs(graph, start, goal):
frontier = deque([start])
explored = set()
while frontier:
  node = frontier.popleft()
  if node == goal:
    return True
  explored.add(node)
  for neighbour in graph[node]:
    if neighbour not in explored:
      frontier.append(neighbour)
return false

graph = {
        'A': ['B', 'C'],
        'B': ['A', 'D'],
        'C': ['A', 'D'],
        'D': ['B', 'C', 'E'],
```

print (bfs (graph, 'A', 'F'))

Output: True

AI 10/04/2025

PROBLEM - SOLVING AND SEARCH TECHNIQUES IN AI

- Types of search

* Uninformed search is also called Blind search;
- These methods don't have any additional information about the states beyond that which is provided in the problem definition Eg Depth-first search (DFS)

Depth First Search: DFS explores as far as possible along its branch before backtracking.

Pseudocode

```
def dfs(start, goal, visited=None)
  if visited is None
    visited = set()
  visited.add(start)
  if start == goal
  return [start]
  for neighbour in graph[start]:
  if neighbour not in visited
  path = dfs(graph, neighbour, goal, visited)
  if path:
  return [start] + path
  return None
```

2* Informed Search (Heuristic-Based Search)

These algorithms uses problem specific knowledge to find solutions.

Key methods, Greedy-Best-first search, A* Search

Greedy-Breath-first search - uses a heuristic to choose the path that appears best.

Heuristic (meta heuristics) are problem solving approaches used in optimization, AI & operation research. They help find (not necessarily optimal) solutions when exact methods are too slow (impractical) (unfeasible). A heuristic is a practical, experience based method designed to solve a problem quickly when traditional methods are either too slow or unable. Examples of heuristics are; Greedy algorithms (knapsack algorithm), Nearest neighbour, Rule of thumb strategies in decision making.

META HEURISTICS: This is a high level strategy that guides heuristic methods to explore the solution space more effective. Unlike heuristic, metaheuristic they are not problem specific & can be applied to a wide range of problems e.g. Evolutionary Algo. (Genetic algos.)

Ant colony, simulated annealing, Tabu search.

II) ~~A*~~ Greedy search algorithm

```
def gbf(graph, start, goal, heuristic)
    frontier = PriorityQueue()
    frontier.put((heuristic[start], start, [start]))
    visited = set()
    while not frontier.empty():
        node, path = frontier.get()
        if node == goal:
            return path
        if node not visited:
            visited.add(node)
            for neighbour in graph[node]:
                frontier.put(heuristic[neighbour]),
                neighbour, path
            frontier.put(heuristic[neighbour]), neighbour, path+[neighb
    return none
```

II A* search – Combines the cost to reach a node & the heuristic estimate, from that node to the goal

Adversarial search – It is used in a turn ~~Plays~~ game player scenario where the players takes turn

Alg- Minimax algorithm, Alpha beta pathing pruning.

Exam Req:

1) Search techniques, Why Search as a concept is important in AI.
2) Different types of search techniques, the possibilities or implementation details.
3) A case study scenario

SYMBOLIC PROBLEM SOLVING

Symbolic Pattern problem solving uses explicit symbol manipulation to solve problem eg Rules, Logic, Search

* Case study of Search 1) Path finding & Robotics

Scenario: Autonomous

Problem: A Robot must find the shortest path from point A & point B to avoid obstacles.

Solution: A* search is used due to its optimality and efficiency.

Symbolic Problem Solving {cont'd}

Roles of Symbolic Manipulation in AI

AI can represent knowledge using symbol and reason to represent rules used for theory proving.

(1) QUEEN PROBLEM

A place queens on a chess board so that no two queens can threaten each other

(2) TOWER OF HANOI – Move disk from source to destination peg following rules:

(1) Only one disk at a time

(2) A large disk cannot be placed on a smaller one

(Tower of Hanoi)

```
def Hanoi(n, source, auxiliary, target):
    if n == 1:
        print("move disk 1")
        return
    Hanoi(n-1, source, target, auxiliary)
    Hanoi(3, 'A', 'B', 'C')
```

DOMAIN ANALYSIS & Knowledge Representation

1) Knowledge representation and reasoning is central to AI because it answers 2 fundamental questions:

(1) How can you represent knowledge about the world in a form that a computer can understand and process.

(2) How can we use knowledge to ~~infer~~ infer new facts or make decision

LOGICAL REPRESENTATION

Propositional logic: This is the simplest form of logic. It can represent facts as true or false logic. It uses connectives like AND

$\wedge$(AND), $\vee$(Or), $\neg$(not), $\rightarrow$(implies), $\leftrightarrow$(iff)

Examples

If it rains, the ground is wet

Rain → Wet

First order logic: First order logic extends propositional logic with quantifiers, variables and relations.

Quantifiers:

∀ (for all)

∃ (there exist)

Examples:

"All human are mortal":

$\forall x$ (Human(x) → Mortal(x))

"Socrates is a human": Human(Socrates)

EXPLORATORY DATA ANALYSIS (LOAN PREDICTION PROBLEM)

* Importing libraries & the Loan prediction data

LOAN PREDICTION

- Your Loan Definition.

Loan is a thing that is borrowed, especially a sum of money that is expected to be paid with interest

Banks get a major part of their profit from loans.

It is crucial for banks to determine whether or not a loan can be approved based on the applicant's profile

- Machine L. code in python will be used to predict whether the candidate's profile is relevant

4 ~~CNN~~, Google Colab, Kaggle

1) describe function of diff. described

COMPILER CONSTRUCTION MAY/02/2025

- Parsing Method

1) Bottom up (1) Top down parsing

4) Bottom up begins from terminals to non-terminal

4) Top down parsing begins from non-terminals to terminals

— We implement parsing using stack method FIFO & LIFO.