Turing Test

→ proposed by Alan Turing - 1950 - his paper "computing Machinery and Intelligence"
→ is a method to evaluate machine ability to exhibit human like intelligence
→ is based on party game called "imitation Game".
→ test involve 3 participant
  ◦ computer ◦ human responder ◦ human interrogator
→ human interrogator communicates with two participant through text based interface without knowing which is human and which is comp.
→ interrogator task is to determine which participant is computer based on their responses to various question.
→ if interrogator is unable to reliably distinguish the computer from the human the machine is said to have passed the test.

Types of AI Agent

① Simple Reflex
→ make decision on the basis of current percept.

② ~~Model Based Reflex~~
→ Based on If-then rules
→ Environment should be fully observable

② model Based Reflex.
→ uses model to keep track of what going on over time.
→ Partially observable environment

③ Goal Based
→ Extension of model Based Reflex
→ makes decision based on goals it want to achieve.
→ eg. GPS that plan the best route to your decision.

④ Utility Based
→ choose action not just to achieve goal to get the best overall result
→ It measure how good each possible action is and pick the one that give the best outcome.

⑤ Learning Based
→ Learn from its experience to improve over time
→ eg. recommendation system that improve its suggestion based on what you like.

8 puzzle problem

→ ~~objective~~ is a classic sliding puzzle where the goal is to move tiles on a 3×3 grid to reach a target configuration.
→ Heuristic fun^n in context to this problem is used to estimate the cost of reaching the goal from a given state.
→ 2 common heuristic function used:
(1) ~~Hamming~~ h1 → counts no. of tiles that are not in their goal position
(2) h2 → sum of the distance of each tile from its goal position.

Steepest Ascent Hill Climbing

→ is a heuristic search algorithm used for solving optimization problem
→ A node of hill climbing algorithm has two component which are state and value
→ The state space landscape is graphical representation of hill climbing algorithm which is showing graph b/t various state of algorithm and objective function.
→ used to find local maxima and minima in a search space.

local maxima: state which is better than its neighbouring state however there is a state which is better than its.
~~global~~ Global maxima: best possible state in state space diagram.
Plateau/flat local maxima: It is a flat region of state space where neighbour have the same value
Ridge: It is region higher than its neighbours but itself has a slope
Current state: region where we are currently present during the search.

Generate & Test	Best first search
→ generate possible sol^n and test each one to see if it meet the goal.	→ prioritize the node based on a heuristic function
→ can be inefficient; tests many sol^n.	→ more efficient, focus on promising path based on heuristic.
→ Not guarantee for optimality	→ guarantee optimality
→ Typically have low memory usage	→ can have high memory usage due to storing all node in priority queue
→ does not uses heuristic	→ Uses heuristic fun^n
→ Uninformed does not use any info. beyound the current state	→ Informed uses heuristic to estimate the cost to reach goal

Date : Page : of

Production System

→ In AI and CS, a production system is a model used for problem solving and decision making.

→ It consist of set of rules (production) and a working memory to store current state of problem.

→ consist of 3 component

① Production Rules – condition-action pairs – defines what action to take when specific condition is met.

② Working memory – Stores the current state of problem

③ Control System (Inference Engine) – manage which rule to apply and in which order.

Problem Reduction using AO*

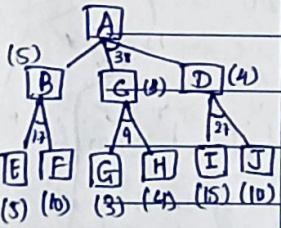
→ technique used to break down complex problem into simple sub problem

→ helps in systematic problem solving by structuring problem as a graph where nodes represent sub problem and edges represent relationship

→ Evalution fun^n.

$f(n) = g(n) + h(n)$ → estimated cost from current node to goal node.

↓ initial cost from node to current node



I: A→B (1+5) = 6 ✓
A→C→D (1+3)+(1+4) = 9

II: B→E→F (5+1)+(10+1) = 17 on hold

III: A→(C,D) = A→C→(G,H) = 9+27 = 36

CryptArithmetic Problem

→ Type of constraint satisfaction problem

→ Constraint → No two letter have same value
- sum of digits must be as shown in problem
- There should be only one carry forward

Digits that can be assigned to a word/alphabet (0–9) range.

Eg

```
  TO
+ GO
-----
 OUT
```

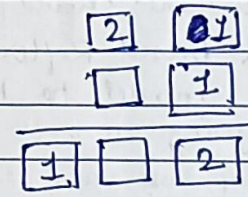
letter	Digit
T	2
O	1
G	8
U	0

left most digit = 1

Possible max. of two number 9+8 = 17

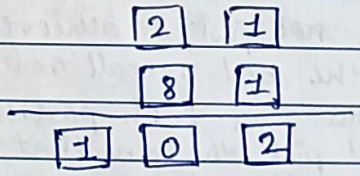
Carry it can't be 0 ∴ O→1

O+O = T
1+1 = T
T = 2



So, 2+G = U+10, since we want two digit number.

Possibility
2+9 = 11 ✗ → becz O=1 we can't assign same value to two alphab
2+8 = 10 ✓



Teacher's Signature :

Prolog declarative nature helps in AI

→ allows developer to express knowledge and rules in more natural way similar to how human reason.

- Represent complex knowledge
  ↳ can represent intricate relationship and concept in structured manner
- Reason and infer
  ↳ system can automatically deduce conclusion based on given facts and rules
- Solve Problem
  ↳ prolog backtracking mechanism allows to explore multiple soln paths and find the best one
- Explain Reasoning
  ↳ system can provide explanation for its conclusion make it easier to understand.

Prolog is primarily designed for deterministic reasoning tasks. It is not inherently designed for probabilistic reasoning which involves uncertainity and stochastic process

However with extensions like ProbLog or PRISM we can handle probabilistic reasoning.

Limitation of traditional prolog on handling probabilistic reasoning

- NLP - understanding ambiguity and context of language often involves probabilistic reasoning
- ML - many ML models rely on probabilistic models to make prediction & decision
- Adaptability - AI system often learn from data and update their knowledge. Without probabilistic reasoning, Prolog can't easily adapt or generalize from uncertain data.

How Prolog Handle uncertainty

- combine prolog with other languages
- use special prolog extension
- try languages designed for probability
- Negation as Failure.

Limitation of Expert System

- Knowledge Limitation : limited to specific domain and struggle with generalization
- Lack of Common Sense : cannot apply reasoning beyond programmed knowledge.
- Inflexibility : difficult to modify or update as knowledge evolve
- High Development Costs : Building and maintaining them can be expensive
- No Learning capabilities : cannot improve from experience unless manually updated

Rule Based ES	Case Based ES
• Knowledge Representation: based on a set of if-then rules.	• stores past cases with soln for future reference
• Apply logical rules to known facts to reach a conclusion	• Solve problem by finding and adapting similar past cases
• Less flexible	• more flexible
• Does not learn; require manual update for new rule.	• learn from new cases, improving as more cases are stored.
• Limited ability to handle uncertainty unless explicitly programmed.	• can handle complex, unstructured problem through case comparison.

NLP (Natural language Processing)

→ is subfield of AI focuses on the interaction b/t computer & human (natural) language.

→ is the backbone of virtual assistant like Siri, Alexa, Google Assistant

→ It enables these assistants to understand and respond to human language in a natural and intuitive way.

→ Key Role : • Speech Recognization • Natural language Understanding • Text generation • Dialogue management

Rule Based NLP

→ uses predefined rules and pattern to process and analyze text.

→ involve creating a set of linguistic rules ~~so~~ that system follows to understand language.

→ Example: Simple chatbots, keyword extractor.

ML Based NLP

→ utilizes machine learning algo to learn from data and improve language understanding.

→ involves training models on large datasets to recognize pattern and make prediction based on new input

→ Example: Sentiment analysis, language translation.

NLP techniques to handle ambiguity

- Contextual analysis - examine words and phrases
- Part of speech Tagging - knowing if word is noun, verb etc.
- Semantic Analysis - knowing how words connect
- World Knowledge - Real world facts
- ML

Syntactic	Semantic
→ ~~purpose is to understand relationship b/t words~~	→ focuses on meaning and context
→ focuses on grammar and structure	→ ~~focuses on~~
→ Types of error detected: Syntax error	→ semantic error
→ Performed first in the process	→ performed after syntactic analysis to check meaning
→ Tools used: parser, compiler	→ Abstract syntax tree (AST), type checker

Sentiment Analysis (Opinion mining)

→ in NLP is the process of identifying and classifying the emotional tone of a text.

→ Aim is to classify the sentiment into categories like +ve, -ve or neutral, or provide a result such as emotion detection (e.g. joy, anger, surprise).

Techniques used for Analysis

1. Dictionary-based method
   → use sentiment lexicons (list of words with ~~sentiment~~ associated ~~list~~ of scores) to determine overall sentiment of text.
2. Machine learning based method
   → train ML models on labeled dataset to classify text into sentiment categories.
3. Hybrid methods
   → combine both dictionary and ML based ~~meth~~ approach for improved accuracy.

Challenges of scaling Prolog Based system

- Computational complexity
- memory consumption - consume lot of memory
- Scalability - difficult to handle large amt. of data
- Integrating with other systems - can be complex due to difference in paradigms and data structure
- Debugging & Maintenance - prolog code can be challenging due to declarative nature and backtracking mechanism. Maintenance large Prolog KB can be time consuming.

Propositional logic	Predicate Logic
→ It is a basic and most widely used Logic.	→ extension of propositional logic covering predicate and quantification
→ has a specific truth value either T or F	→ truth value depends on variable value.
→ is more generalized representation	→ specialized
→ cannot deal with sets of entities (SOE)	→ can deal with SOE with the help of quantifies

Modus Ponen

→ also known as law of Detachment

→ simple rule in logic that help us to make conclusion based on if then statement.

Eg. Premises 1 : If P, then Q $(P \rightarrow Q)$
Premises 2 : P (the antecedent is true)
Conclusion : Therefore, Q (the consequent is true).

Notation for Modus ponen:

$$\frac{P \rightarrow Q, P}{\therefore Q}$$

→

P	Q	$P \rightarrow Q$	$P \wedge (P \rightarrow Q)$
0	0	0	0
0	1	1	1
1	0	0	1
1	1	1	1

Knowledge Representation

- Logical representation
- Semantic Network
- Frames
- Production Rule

GamePlaying

→ imp. Domain in AI

→ don't require much knowledge; the only knowledge we need to provide is the rules, legal moves and cond^n of winning and lossing the game.

- Search Algorithm
- Reinforcement Learning
- Deep learning
- Transfer Learning
- multi Agent System.

modeling Two-Player Zerso sum game

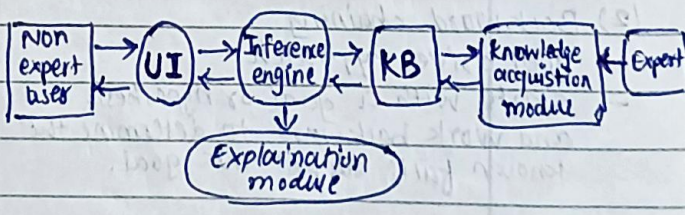
→ In AI, these games can be model using game theory, a branch of mathematics that analyzes ~~stragies~~ strategies for decision making.

- min max Algorithm
- Playoff matrix
- Nash Equilibrium
- Reinforcement learning

Expert System

→ is a subset of AI that simulate the decision making ability of a human expert.

→ uses knowledge stored in the KB to solve problem that would usually require a human expert thus preserving a human expert knowledge in its KB.



Knowledge Base in expert System

→ core component
→ serving as its repository of expertise
→ contain factual info, rules, heuristics that system uses to reason & make decision.

Creation of KB

- Define domain - clearly outline the specific area or field that the expert system will cover.
- Gather Knowledge - consult with domain expert to extract critical knowledge and insights, Research existing document, literature
- Organize knowledge : use structured format like production rules, frames etc.
- Implement Inference Engine : develop or use forward or backward chaining for decision making.
- Validate & Test : Ensure accuracy with expert validation and scenario based testing
- Maintain & Update : Continuously refine the system as knowledge evolves.

Inference Engine in expert System

→ core component
→ responsible for applying logical reasoning to KB to derive conclusion or solve problem
→ engine analyze i/p data or facts provided by user. It uses set of rules (if-then stmt) from the KB to match i/p data with possible conclusion. Based on this matches, it deduce new info, or recommend actions.
→ There are two primary reasoning techniques

(1) Forward chaining
→ data driven approach
→ start with known facts and applies inference rule to derive new fact until a goal is reached.

(2) Backward chaining
→ goal driven approach
→ starts with a goal or hypothesis and work backward to determine the known facts support this goal.

CSP

→ goal is to find a soln to a problem defined by a set of variables, each of which satisfy certain constraint
→ are widely used in AI, particularly for problems like scheduling, planning and resource allocation
→ Example of CSP : sudoku puzzle, map coloring, N-Queen problem

Constraint Propogation Algorithm

→ technique used in CSP which systematically reduces the possible value for each variables before or during search for a solution.

Procedural Knowledge	Declarative Knowledge
• also known as imperative knowledge	• functional knowledge
• can't easily communicate	• easily communicable
• represented by set of rules	• Production System
• Validation not easy	• quite easy
• Debugging not easy	• quite easy
• is process oriented	• data oriented

KBA

→ are those agent who have capability of maintaining internal state of knowledge, reason over that knowledge, update their knowledge after observation and take action
→ are composed of 2 main parts :
• KB (central component)
• Inference system (apply logical rules to KB to deduce new info) (generate new facts so that agent can update KB)