Artificial Intelligence

Unit –I

B.Tech. V Semester
CSE C & D

By
Dr. Loshma Gunisetti,
Associate Professor
CSE Department

Textbook:

1. Artificial Intelligence, Saroj Kaushik, 1st Edition, Cengage Learning

Reference Books:

1. Artificial Intelligence, Elaine Rich, Kevin Knight, Shivashankar B Nair, 3rd Edition, Tata McGraw Hill Education Private Limited., 2009
2. Artificial Intelligence- A modern Approach, 3rd Edition, Stuart Russel, Peter Norvig, Pearson Education

Course Outcomes

After successful completion of this course, the student will be able to:

CO	Course Outcomes	Knowledge Level
1	**Illustrate the concept of Intelligent systems and current trends in AI**	**K2**
2	Apply Problem solving, Problem reduction and Game Playing techniques in AI	K3
3	Illustrate the Logic concepts in AI	K2
4	Explain the Knowledge Representation techniques in AI	K2
5	Describe Expert Systems and their applications	K2
6	Illustrate Uncertainty Measures	K2

Introduction to Artificial Intelligence (AI)

Definitions of AI:

- "The science and engineering of making intelligent machines" *(1956, John McCarthy)*
- "The study of how to make computers do things at which, at the moment , people are better" *(1991,Rich and Knight)*
- "The branch of computer science that is concerned with the automation of intelligent behavior" (*1993, Luger and Stubblefield*)

Introduction to Artificial Intelligence (AI)

Definitions of AI:

- The theory and development of computer systems to perform tasks normally requiring human intelligence, such as visual perception, speech recognition, decision-making, and translation between languages. *(English Oxford Living Dictionary)*
- A branch of computer science dealing with the simulation of intelligent behavior in computers *(Merriam-Webster Dictionary)*

History of AI

- Initially Artificial Intelligence was called as Machine Intelligence
- John McCarthy organized a conference on Machine Intelligence in 1956 and coined the term “Artificial Intelligence”
- He was an American computer scientist, pioneer and inventor
- He is known as the father of Artificial Intelligence (AI)
- In 1957, first version of a new program named as General Problem Solver(GPS) was developed by Newell and Simon
- GPS was capable of solving to some extent the problems requiring common sense.

History of AI

- In 1958, John McCarthy developed **LIS**t **P**rocessing Language (**LISP**)
- LISP was adopted as the language of choice by most AI developers
- Marvin Minsky of MIT demonstrated that computer programs could solve spatial and logic problems when confined to a specific domain
- In 1960, a program named **STUDENT** was developed which could solve algebra story problems.

History of AI

- Fuzzy Set and Logic was developed by L Zadeh in 1960 that had the unique ability to make decisions under uncertain conditions
- Neural Networks were considered as possible ways of achieving Artificial Intelligence
- **MACLISP** language was used to develop a program called SHRDLU by Terry Winograd at MIT
- **PROLOG** language was proposed by R Kowalski, Imperial College, London

History of AI

- In 1970, Expert Systems were developed to predict the probability of a solution under a set of conditions
- An ***expert system*** is a program that uses logical rules that are derived from the knowledge of experts to answer the question or solve problems about a specific domain of knowledge
- **Expert Systems** were used for
  - *forecasting the stock market*
  - *aiding doctors with ability to diagnose disease*
  - *instructing miners to promising mineral locations*

Intelligent Systems

- AI is the study of making computers do things intelligently.
- AI is a combination of computer science, physiology and philosophy
- AI programs must have capability and characteristics of intelligence such as *learning, reasoning, inferencing, perceiving and comprehending information*

Intelligent Systems

There are two views of **AI goals**:

- AI is about duplicating what the human brain does (Cognitive science)
- AI is about duplicating what the human brain should do, that is, doing things logically or rationally

Intelligent Systems

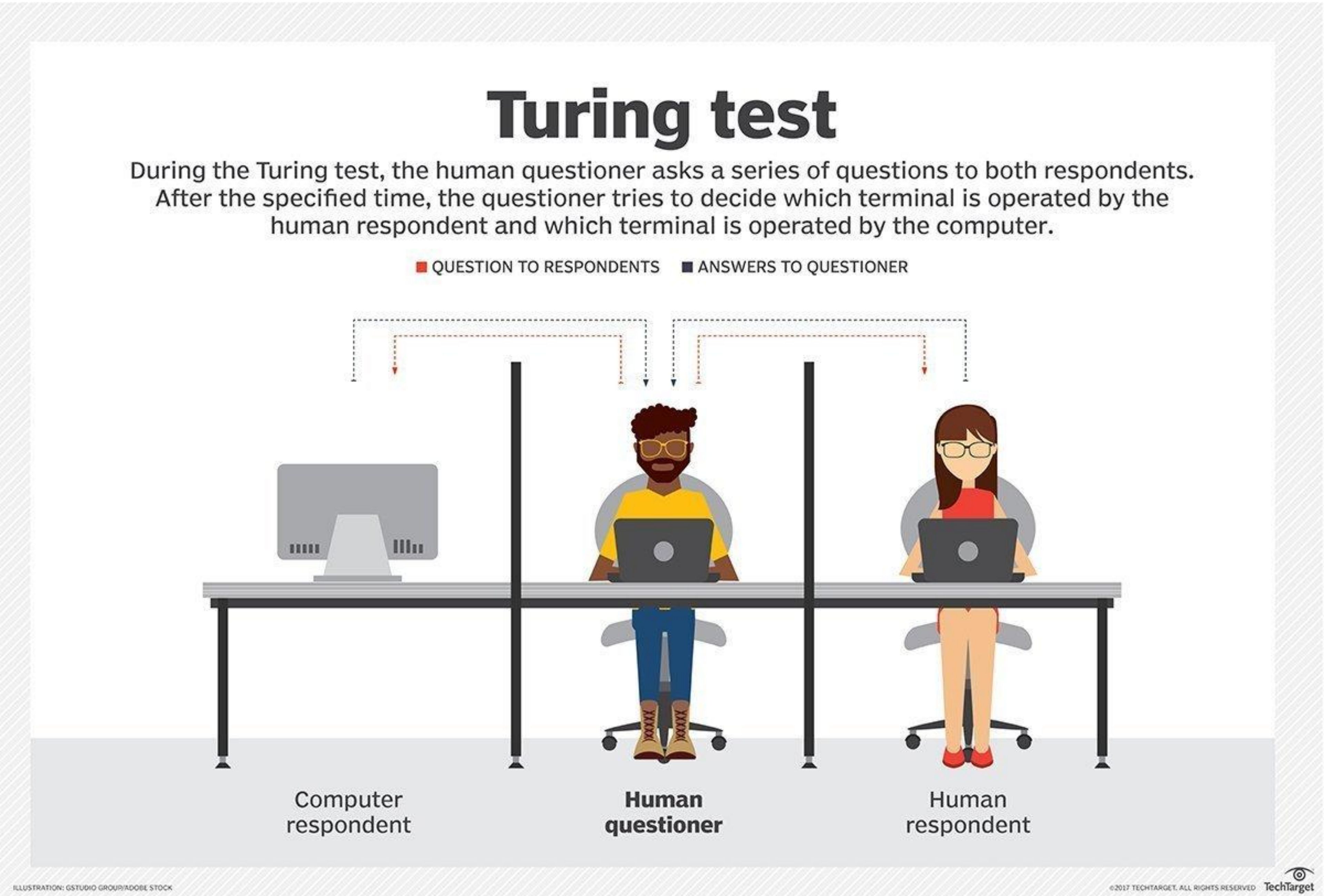
- There a lot of challenges in building systems that mimic the behavior of the human brain which is made up of a billion of neurons
- In order to classify systems as intelligent , it is necessary to define intelligence
- The earliest best method to gauge the intelligence of a system was the **Turing test** proposed by Alan Turing in 1950
- He said that a system is called intelligent if it has an ability to pass the Turing test.

Intelligent Systems

- **Turing test** is informally defined as :

  A system is said to have passed the Turing test, if a human questioner is unable to determine from repeated questions of any kind, whether he or she is talking to another person or to a machine/system

- The very first intelligent system named **Eliza** passed the Turing test which was written by Joseph Weizenbaum during 1964 to 1966 at MIT


Turing test
During the Turing test, the human questioner asks a series of questions to both respondents. After the specified time, the questioner tries to decide which terminal is operated by the human respondent and which terminal is operated by the computer.
QUESTION TO RESPONDENTS
ANSWERS TO QUESTIONER
Computer respondent
Human questioner
Human respondent
ILLUSTRATION: GSTUDIO GROUP/ADOBE STOCK
©2017 TECHTARGET. ALL RIGHTS RESERVED
TechTarget

Eliza: Intelligent System

- Eliza was a program designed to imitate a therapist
- It conversed with the user in English.
- It was able to converse about any subject using information stored in data banks
- It picked up speech patterns from user's questions and provided responses using those patterns
- Sample dialogue between Eliza and a teenage girl is as follows:

Passage of Eliza Conversation

Eliza: Intelligent System

Main characteristics of Eliza:

- **Simulation of Intelligence:** Simulated intelligent behavior quite effectively by recognizing key words and phrases
- **Quality of response:** Success depended on the fact that the user had a restricted idea of expected response from the system
- **Coherence:** Initial versions were based entirely on the current input and no context information was used. Intelligence depended on the coherence(focus) of the conversation as judged by the user
- **Semantics:** No semantic representation of the content of user input or reply, it does not have intelligence of understanding of what we are saying. It imitates human conversation style, hence it passed Turing test

Categorization of Intelligent Systems

Four categories of Intelligent Systems are:

- **System that thinks like humans:** Requires cognitive modelling approaches, functioning of the brain should be known
- **System that acts like humans:** Overall behavior should be human-like which could be achieved by observation
- **System that thinks rationally:** Relies on logic than humans to measure correctness
- **System that acts rationally:** Doing the right thing, even though the method could be illogical

Intelligence

- Intelligence can be defined as that property of mind which encompasses many related mental abilities such as:
  - Reason and draw meaningful conclusions
  - Plan sequences of actions to complete a goal
  - Solve problems
  - Think abstractly
  - Comprehend ideas and help computers to communicate in Natural Languages
  - Store knowledge provided before or during interrogation
  - Learn new ideas from environment and new circumstances
  - Offer advice based on rules and situations
  - Learn new concepts and tasks that require high levels of intelligence

Components of AI Program

Any AI program should have :

- **Knowledge base-** AI programs should be learning in nature and update their knowledge accordingly . Knowledge base generally consists of facts and rules and has the following characteristics:
  - It is voluminous in nature and requires proper structuring
  - It may be incomplete and imprecise
  - It may be dynamic and keep on changing
- **Control Strategy-** It determines which rule to be applied
- **Inference Mechanism-** It requires search through knowledge base and derives new knowledge using the existing knowledge with the help of inference rules

Foundations of AI

As AI is interdisciplinary in nature, foundation of AI is in various fields such as:

- Mathematics
- Neuroscience
- Control Theory
- Linguistics

Foundations of AI

- **Mathematics**

AI systems use formal logic methods and Boolean Logic, probability theory, uncertainty etc.

- **Neuroscience**

This science of medicine helps in identifying the functioning of brains. Accurate sensors can be monitored to correlate brain activity to human thought

Foundations of AI

- **Control Theory**

  Machines can modify their behavior in response to the environment . For eg. Water flow regulator

- **Linguistics**

  Speech demonstrates a lot of human intelligence. Languages and thoughts are believed to be tightly intertwined.

Applications of AI

AI has applications in almost all areas of real life applications :

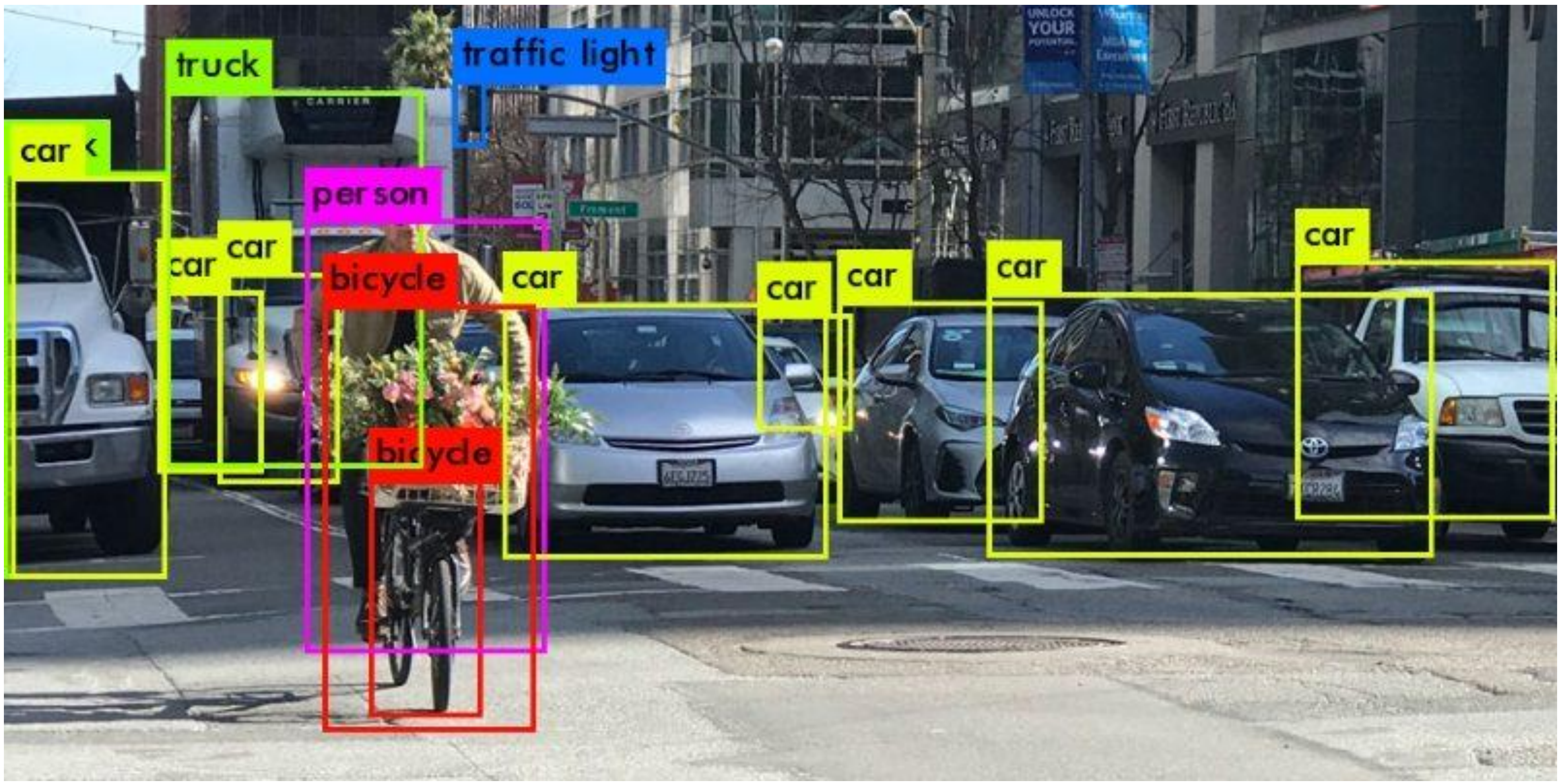
- **Business** : financial strategies, give advice
- **Engineering**: check design, offer suggestions to create new product, expert systems for all engineering applications
- **Manufacturing**: assembly, inspection, and maintenance
- **Medicine**: monitoring, diagnosing and prescribing
- **Education**: teaching

Applications of AI

- **Fraud detection :** With the increase in the number of transactions happening across the globe, the threat of financial fraud has increased too. AI based software can be used to monitor and detect any deviations from the normal pattern in transactions

Applications of AI

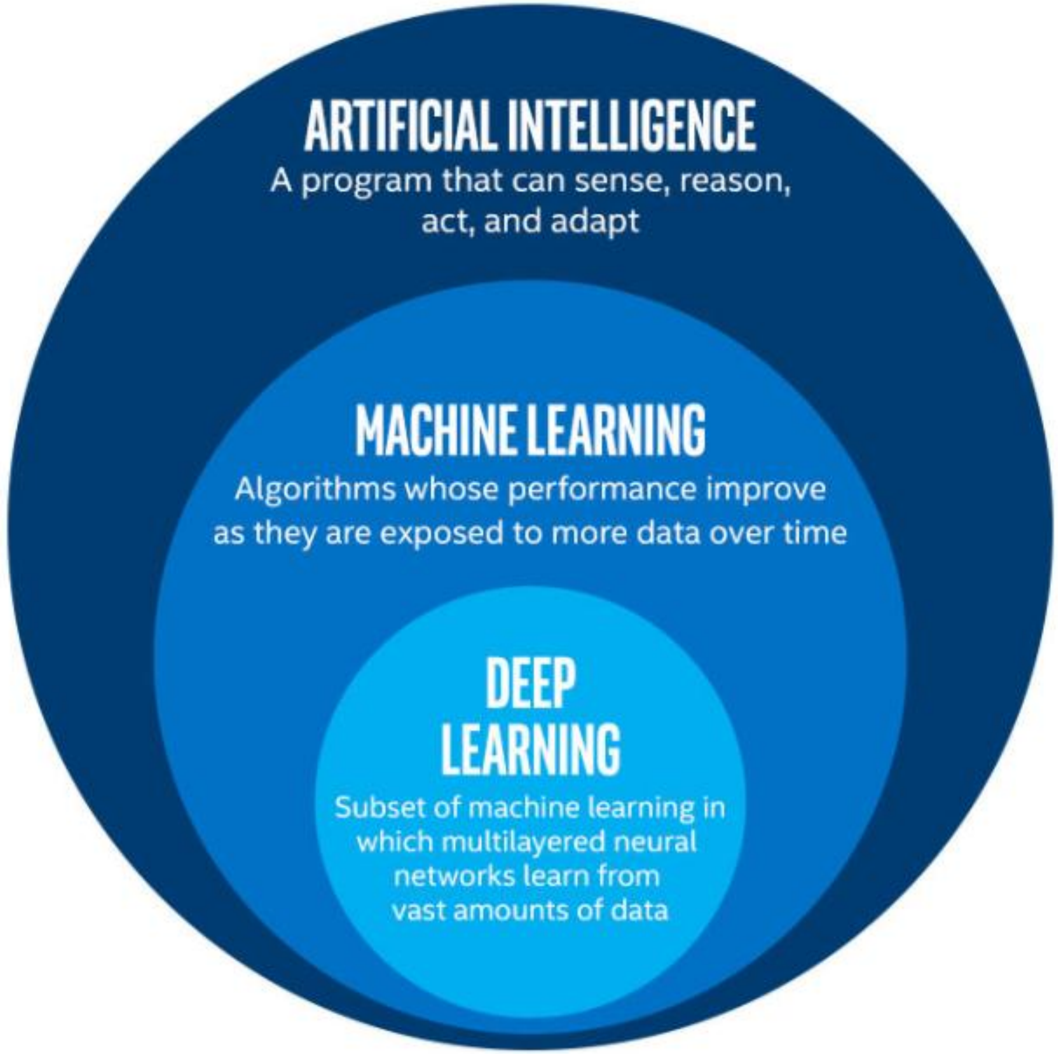
- **Object Identification –** For Self-driving cars



Applications of AI

- **Space shuttle scheduling-** Automated planning and scheduling of space shuttles

- **Information retrieval -**Retrieve documents with information that is relevant to user's information need and helps the user complete a task , for eg. any search engine


ARTIFICIAL INTELLIGENCE
A program that can sense, reason, act, and adapt
MACHINE LEARNING
Algorithms whose performance improve as they are exposed to more data over time
DEEP LEARNING
Subset of machine learning in which multilayered neural networks learn from vast amounts of data

AI vs. ML vs. DL

Applications of AI in transportation

1 : Google's AI-Powered Predictions

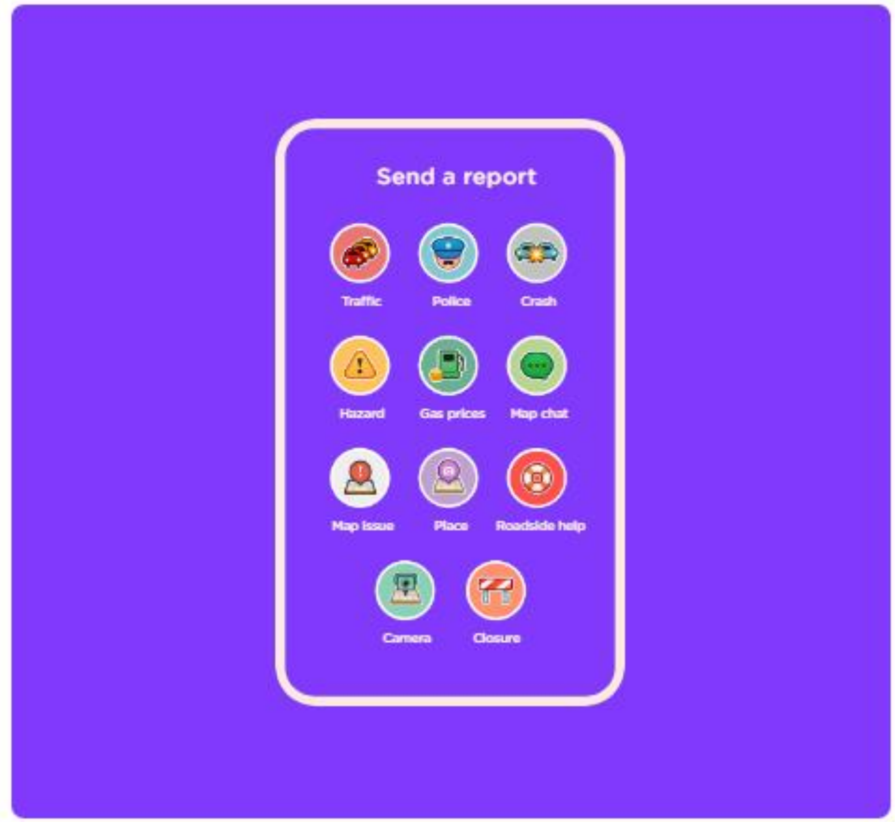
Using anonymized location data from smart phones, Google Maps (Maps) can analyze the speed of movement of traffic at any given time. And, with its acquisition of crowdsourced traffic app **Waze** in 2013, Maps can more easily incorporate user-reported traffic incidents like construction and accidents. Access to vast amounts of data being fed to its proprietary algorithms means Maps can reduce commutes by suggesting the fastest routes to and from work.

Waze is a community-driven GPS and navigational app that guides you through the shortest route possible while driving.

Source : https://emerj.com/ai-sector-overviews/everyday-examples-of-ai/

Applications of AI in transportation

Waze helps riders and drivers get where they're going—faster, smoother, safer, and happier—while working to beat traffic. User knows what is ahead with real-time help from other drivers.

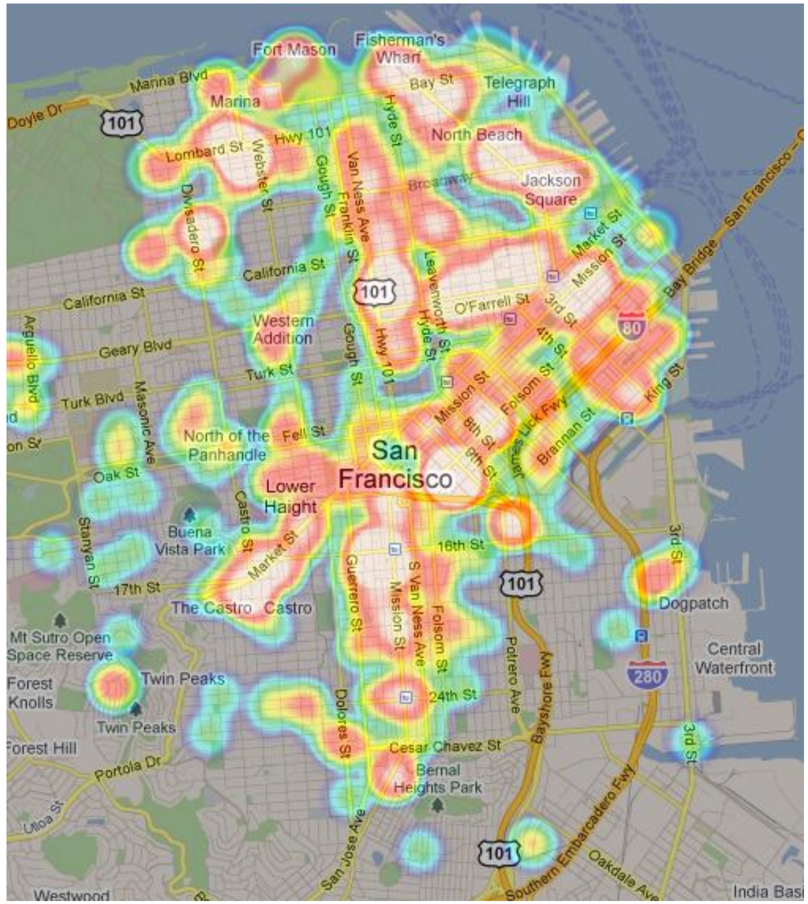


Applications of AI in transportation

2 : Ridesharing Apps Like Uber and Lyft

Uber uses Machine Learning (sub-area of AI) to predict rider demand to ensure that “surge pricing”(short periods of sharp price increases to decrease rider demand and increase driver supply) will no longer be necessary. Uber uses machine learning for ETAs (Estimated Time of Arrival) for rides, estimated meal delivery times on UberEATS, computing optimal pickup locations, as well as for fraud detection

Applications of AI in transportation



Uber Heat Map

Applications of AI in transportation

3 : Commercial Flights Use an AI Autopilot

AI autopilots in commercial airlines is a surprisingly early use of AI technology that dates as far back as 1914. The New York Times reports that the average flight of a Boeing plane involves only seven minutes of human-steered flight, which is typically reserved only for takeoff and landing.

Applications of AI in transportation

In the future, AI will shorten your commute even further via self-driving cars that result in up to **90% fewer accidents**, more efficient ride sharing to reduce the number of cars on the road by up to 75%, and smart traffic lights that reduce wait times by 40% and overall travel time by 26% in a pilot study.

Applications of AI in Email

1 : Spam Filters

Spam filters must continuously learn from a variety of signals, such as the words in the message, message metadata (where it's sent from, who sent it, etc.). **Through the use of machine learning algorithms, Gmail successfully filters 99.9% of spam**

2 : Smart Email Categorization

Gmail uses Smart Email Categorization approach to categorize your emails into primary, social, and promotion inboxes, as well as labeling emails as important

Applications of AI in Grading and Assessment

1 : Plagiarism Checkers

Many researchers and college students use Turnitin, a popular tool used by instructors to analyze students' writing for plagiarism. Historically, plagiarism detection for regular text (essays, books, etc.) relies on a having a massive database of reference materials to compare to the student text; however, **Machine Learning can help detect the plagiarizing of sources that are not located within the database, such as sources in foreign languages or older sources that have not been digitized**.

Applications of AI in Grading and Assessment

2 : Robo-readers

Essay grading is very labor intensive, which has encouraged researchers and companies to build essay-grading AIs. The Graduate Record Exam (GRE), the primary test used for graduate school, grades essays using one human reader and one robo-reader called e-Rater.

Applications of AI in Grading and Assessment

2 : Robo-readers

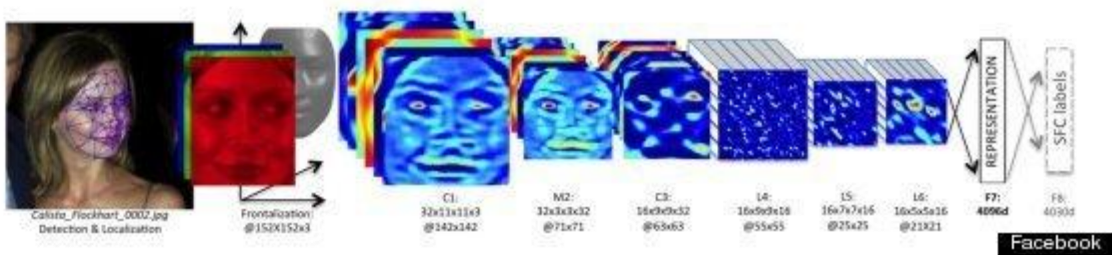
Essay grading is very labor intensive, which has encouraged researchers and companies to build essay-grading AIs. The Graduate Record Exam (GRE), the primary test used for graduate school, grades essays using one human reader and one robo-reader called e-Rater.

Applications of AI: Automotive Industry

AI in Banking/Personal Finance

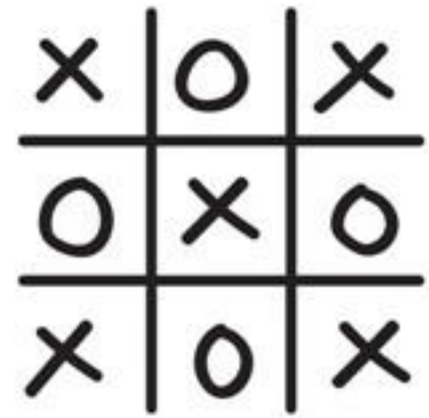
Mobile Cheque Deposit

AI in Social Networking



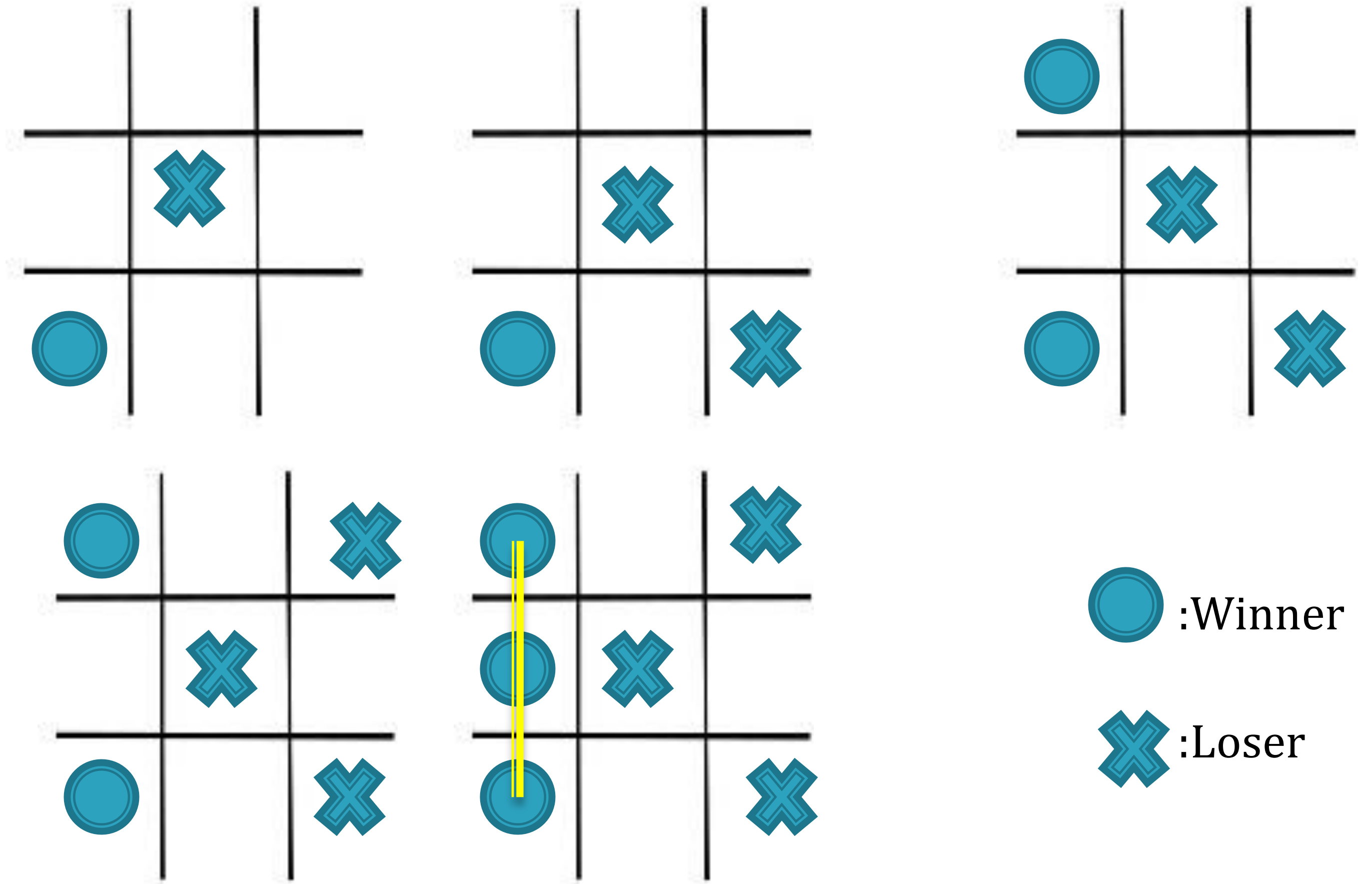
Facebook's facial recognition

Tic-Tac-Toe Game Playing



- Tic-Tac-Toe is a two-player game
- Grid size is **3 X 3**
- One player marking **X** and another marking **O**
- Player who succeeds in placing three respective marks in any horizontal, vertical or diagonal row wins the game.
- Objective is to write a program using which the computer never loses

Tic-Tac-Toe Game Playing



Tic-Tac-Toe Game Playing : Approach 1

1	2	3
4	5	6
7	8	9

Tic –Tac- Toe Board

0- representing blank position

1- indicating X player move

2- indicating O player move

Tic-Tac-Toe Game Playing : Approach 1

- This approach uses a program using a move table consisting of vector of 3^9 (19683) elements

Index	**Current Board Position** 1 2 3 4 5 6 7 8 9	**New Board Position** 1 2 3 4 5 6 7 8 9
0	0 0 0 0 0 0 0 0 0	0 0 0 0 1 0 0 0 0
1	0 0 0 0 0 0 0 0 1	0 2 0 0 0 0 0 0 1
2	0 0 0 0 0 0 0 0 2	0 0 0 1 0 0 0 0 2
3	0 0 0 0 0 0 0 1 0	0 0 2 0 0 0 0 1 0
	. . . .	

Table 1: Move Table

Tic-Tac-Toe Game Playing : Approach 1

1 0	2 0	3 0
4 0	5 0	6 0
7 0	8 0	9 0

Case 1 →

1 0	2 0	3 0
4 0	5 1	6 0
7 0	8 0	9 0

---

1 0	2 0	3 0
4 0	5 0	6 0
7 0	8 0	9 1

Case 2 →

1 0	2 2	3 0
4 0	5 0	6 0
7 0	8 0	9 1

Tic-Tac-Toe Game Playing : Approach 1

- The entries of a Move table are carefully designed manually, in advance, keeping in mind that the **computer should never lose**
- All possible board positions are stored in **Current Board Position** column along with its corresponding next best possible board position in **New Board Position** Column.
- Once the table is designed, the computer program has to simply do the **table lookup**

Tic-Tac-Toe Game Playing : Approach 1

Algorithm:

1. View the vector (board) as a ternary number
2. Get an index by converting this vector to its corresponding decimal number
3. Get the vector from **New Board Position** stored at the index. The vector thus selected represents the way the board will look after the move that should be made
4. So set board position equal to that vector

Tic-Tac-Toe Game Playing : Approach 1

Advantage:

- Efficient with respect to time

Disadvantages:

- Lot of memory requirement to store the move table
- To specify entries in move table manually, lot of effort is required
- Creating move table is highly error prone as data to be entered is highly voluminous
- This approach cannot be extended to 3D tic-tac-toe, as 3^{27} board positions are to be stored
- This program is not intelligent at all as it does not meet any of AI requirements

Tic-Tac-Toe Game Playing : Approach 2

1	2	3
4	5	6
7	8	9

Tic –Tac- Toe Board

X - whoever plays first move (human / computer)
O - whoever plays second move (computer/ human)

2 - represents blank position
3 - indicates X player move
5 - indicates O player move

Tic-Tac-Toe Game Playing : Approach 2

Three sub procedures are used:

Go(n) : Using this function, the computer can make a move in square n

Make_2 : This function helps the computer to make valid 2 move i.e. valid two in any row, column or diagonal

PossWin(P) : If player P can win in the next move then it returns the index(from 1 to 9) of the square that constitutes a winning move, otherwise it returns 0

Tic-Tac-Toe Game Playing : Approach 2

If PossWin(P) =0, then P cannot win. Find whether opponent can win.

If so then block it. This can be achieved as follows:

- If (3*3*2 =18) then X player can win as there is one blank square in row, column or diagonal
- If (5*5*2 =50) then O player can win

Tic-Tac-Toe Game Playing : Approach 2

If PossWin(P) =0, then P cannot win. Find whether opponent can win.

If so then block it. This can be achieved as follows:

- If (3*3*2 =18) then X player can win as there is one blank square in row, column or diagonal
- If (5*5*2 =50) then O player can win

(C plays X, H plays O)
1 move: Go(5) /Go(1)
3 move: If B[9] is blank, then Go(9) else *{make 2}* Go(3)
5 move: *{By now both have played 2 moves} :* If PossWin(X) then *{X wins}* Go(PossWin(X)) else if PossWin(O) *{block O}* then Go(PossWin(O)) else if B[7] is blank then Go(7) else Go(3)
7 & 9 moves: *{By now human (playing O) has played 3 chances} :* If PossWin(X) then *{X wins}* Go(PossWin(X)) else *{block O}* if PossWin(O) then Go(PossWin(O)) else Go(Anywhere)

Table 2a: Rules for Nine moves [*{}-Comments*]

(H plays X, C plays O)
2 move: If B[5] is blank, then Go(5) else Go(1)
4 move: *{By now human (playing X) has played 2 moves} :* If PossWin(X) then *{block X}* Go(PossWin(X)) else *{make 2}* Go(Make_2)
6 move: *{By now computer has played 2 moves} :* If PossWin(O) then *{O wins}* Go(PossWin(O)) else if PossWin(X) *{block X}* then Go(PossWin(X)) else Go(Make_2)
8 move: *{By now computer has played 3 chances} :* If PossWin(O) then *{O wins}* Go(PossWin(O)) else *{block X}* if PossWin(X) then Go(PossWin(X)) else Go(Anywhere)

Table 2b : Rules for Nine moves [*{}-Comments*]

Tic-Tac-Toe Game Playing : Approach 2

Advantages:

- More memory efficient
- Easier to understand as complete strategy has been determined in advance

Disadvantages:

- Not as efficient as first one with respect to time. Several conditions are checked before each move
- Cannot generalize to 3-D Tic –Tac-Toe

Tic-Tac-Toe Game Playing : Approach 3

8	**1**	**6**
3	**5**	**7**
4	**9**	**2**

Magic Square of order 3

- In this approach, we choose board position to be a magic square of order 3
- The magic square of order n consists of n^2 distinct numbers
- Sum for each row, column, diagonal must be $\mathbf{n[(n^2+1)/2]}$ **i.e. 15** in this case

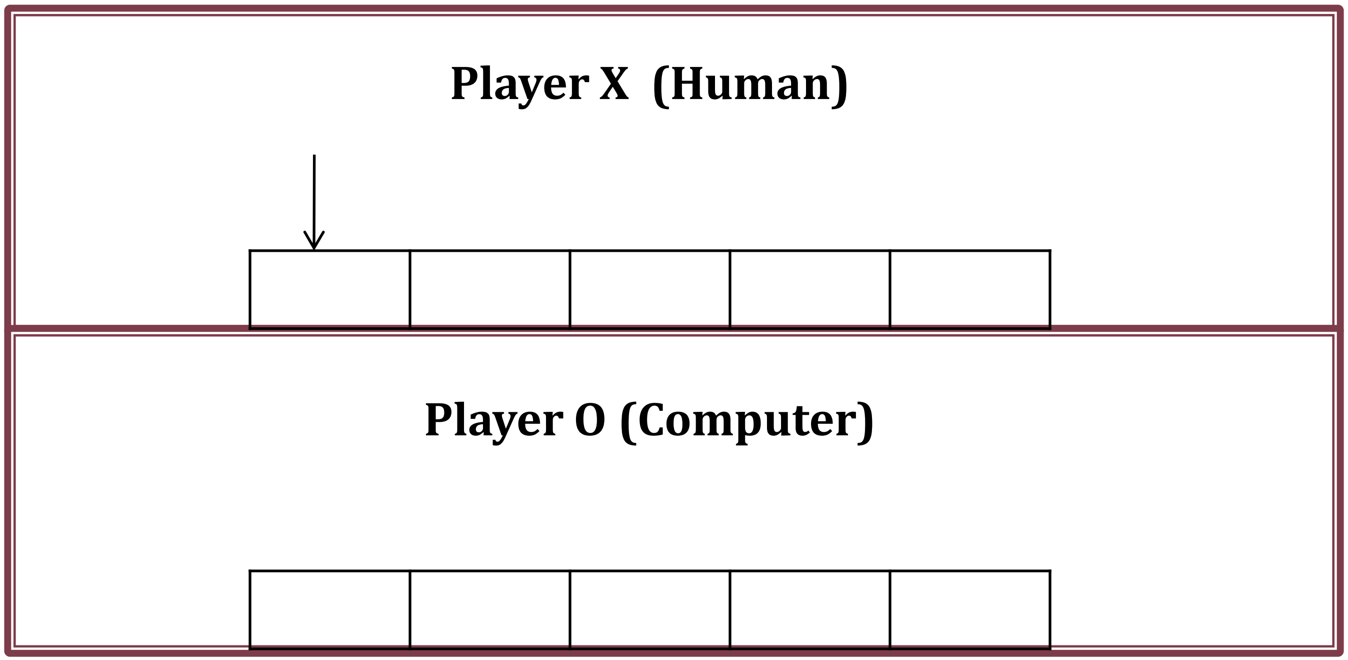
Tic-Tac-Toe Game Playing : Approach 3

- In this approach, we maintain a list of the blocks played by each player, each block is identified by its number
- First few moves are fixed as in Approach 2

Tic-Tac-Toe Game Playing : Approach 3

- Winning strategy for the computer is as follows:
  - Each pair of blocks a player owns is considered
  - Difference D between 15 and the sum of the two blocks is computed
    - If D <0 or D >9, then these two blocks are not collinear and so can be ignored: otherwise if the block representing difference is blank (i.e. not in either list) then player can move in that block
  - This strategy will produce a possible win for a player

Tic-Tac-Toe Game Playing : Approach 3



Initially both lists are empty

Tic-Tac-Toe Game Playing : Approach 3

Player X (Human)

8	1	4	2	

Player O (Computer)

5	6	3		

Status of both lists after Seventh Move

Tic-Tac-Toe Game Playing : Approach 3

Assume that human(H) is the first player(X), computer(C) is second player(O)

- **Turn 1** : Suppose H plays in the eighth block
- **Turn 2 :** C plays in fifth block (fixed move, refer table 2b)
- **Turn 3 :** H plays in first block
- **Turn 4 :** C checks if H can win or not
  - ➢ Compute sum of blocks played by H
    - ▪ S = 8+1 = 9
    - ▪ Compute D =15-9 =6
  - ➢ The sixth block is a winning block for H and not there on either list. So C blocks it and plays in sixth block. The sixth block is recorded in the list of computer
- **Turn 5 :** H plays in fourth block

[contd.]

Tic-Tac-Toe Game Playing : Approach 3

- **Turn 6 :** C checks if C can win as follows:
  - ➢ Compute sum of blocks played by C
    - ▪ S = 5+6 = 11
    - ▪ Compute D =15-11 = 4; Discard this block as it already exists in X list
  - ➢ Now C checks whether H can win
    - ✓ Compute sum of pair of square from list of H which have not been used earlier
      - ▪ S = 8 + 4 = 12
      - ▪ Compute D = 15-12 = 3
    - ✓ Block 3 is free, so C plays in this block. The third block is recorded in the list of computer .

[contd.]

Tic-Tac-Toe Game Playing : Approach 3

- **Turn 7 :** If H plays in second or ninth block, the computer wins. Let us assume that H plays in second block
- **Turn 8 :** C checks if it can win as follows:
  - Compute sum of blocks played by C which has not been used earlier
    - S = 5+3 = 8
    - Compute D =15-8 = 7
  - Block 7 is free, so C plays in seventh block and wins the game
- If H plays in seventh block at its **Turn 7**, then there is a draw.

Tic-Tac-Toe Game Playing : Approach 3

Advantages:

- Can be extended to 3-D Tic –Tac-Toe
- Can be extended to games complicated than Tic-Tac-Toe

Disadvantages:

- Will require much more time than other two approaches as it must search a tree representing all possible move sequences before making each move

3D Tic-Tac-Toe

8	24	10
12	7	23
22	11	9

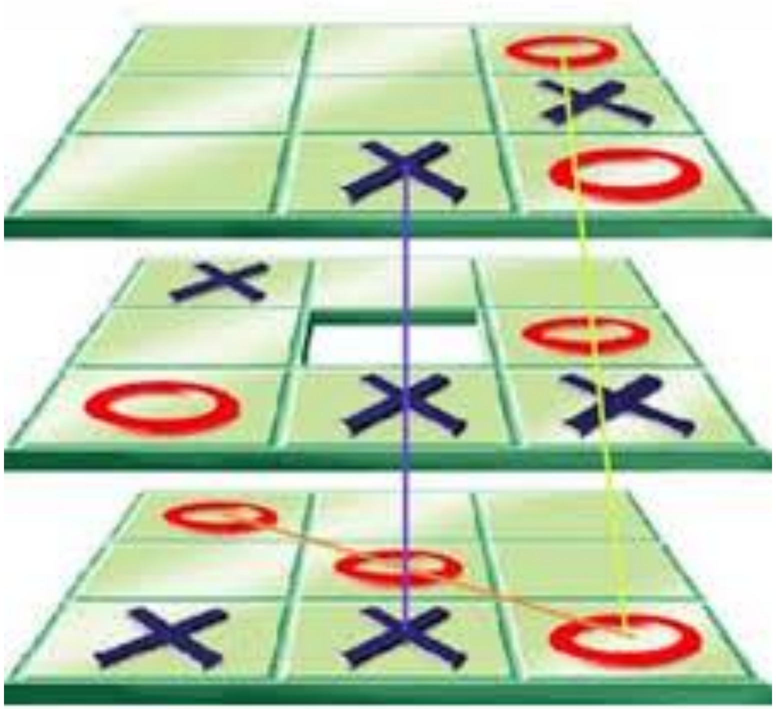
15	1	26
25	14	3
2	27	13

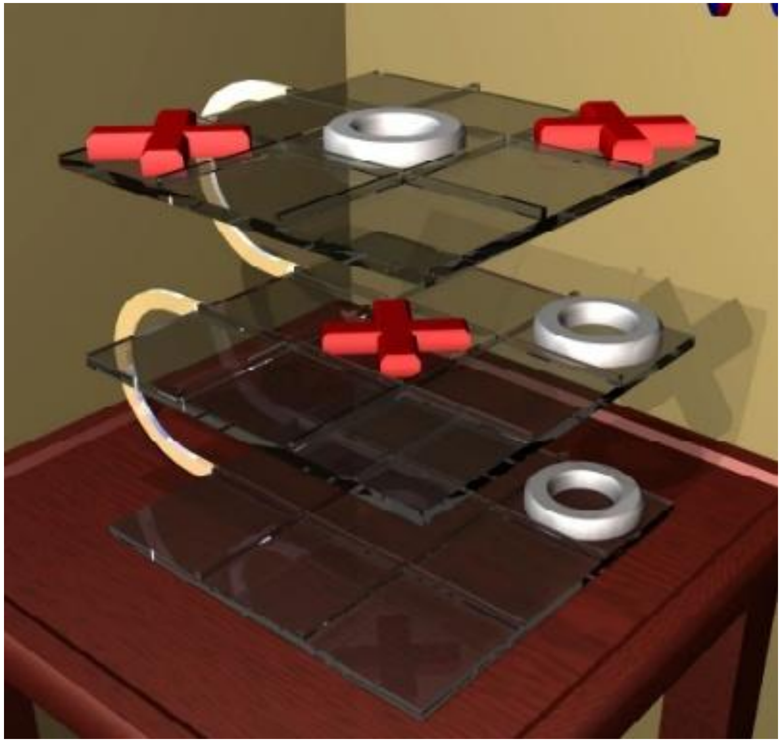
19	17	6
5	21	16
18	4	20

Magic Cube

- Magic Cube of order ***n*** has a magic constant equal to $\mathbf{n[(n^3+1)/2]}$
- Here n**=3,** hence the magic constant is **42**
- Sum of numbers on each row, each column on six outer surfaces of a cube, each row, each column and two diagonals of middle grid and the four main space diagonals is equal to **42**

3D Tic-Tac-Toe Sample Board Layouts





Current Trends in AI

- **Hard Computing** algorithms, also termed as the conventional algorithms, follow mathematical methodologies strictly which make it inefficient to solve real world problems by taking more computation time.
- The conventional algorithms require exact input data, use a precise methodology and generate a precise output which make it a crisp system.
- It fails when the input is not exact.
- **Examples** of conventional algorithms are merge sort, quick sort, binary search, greedy algorithm, dynamic programming etc. which are deterministic.

Current Trends in AI

- **Soft Computing** consists of numerous techniques that study the biological processes such as reasoning, genetic evolution, survival of the creatures and human nervous system. SC is an umbrella term to thoroughly study the simulation of reasoning, human nervous system and evolution in different fields:
  - Neural Networks
  - Fuzzy Logic
  - Evolutionary Algorithms
  - Swarm Intelligence etc.

Hard computing	Soft Computing
Conventional computing requires a precisely stated analytical model	It is tolerant of imprecision
Often requires a lot of computation time	Can solve some real world problems in reasonably less time
Not suitable for real world problems for which ideal model is not present	Suitable for real world problems
It requires full truth	Can work with partial truth
It is precise and accurate	Imprecise
High cost for solution	Low cost for solution

Hard Computing vs. Soft Computing

Current Trends in AI

- **Example:** If we need to find out whether Bob is honest.
  - A hard computing algorithm would give an answer that is Yes or No. (1 or 0 in binary)
  - Soft Computing technique (Fuzzy Logic) would give an answer with membership degree such as extremely honest (1), very honest (0.85), sometimes honest (0.35), extremely dishonest (0.00), like a human.

Current Trends in AI- Soft Computing

- **Neural Network** is a network of artificial neurons, inspired by biological network of neurons, that uses mathematical models as information processing units to discover patterns in data which is too complex to notice by humans.

- **Fuzzy Logic** is a technique that understands the vagueness of a solution and presents the solution with a degree of vagueness which is practical to human decision. It is widely applied in several applications of Artificial Intelligence for reasoning.

Current Trends in AI- Soft Computing

- **Evolutionary algorithms** are the algorithms that are based on the evolution of the species; in general they are based on the main evolutionary theory of Charles Darwin.

  For eg. **Genetic algorithms** were developed mainly by emulating the nature and behaviour of biological chromosome.

  - Genetic algorithms are favoured for search problems which require the identification of a global optimal solution

Current Trends in AI- Soft Computing

- **Swarm Intelligence (SI)** is a type of AI based on the collective behaviour of decentralized, self-organized systems.
- Insects like ants, bees, wasps and termites are able to solve complex problems emerging in their daily lives by mutual cooperation. This emergent behaviour of self organization by a group of insects is known as Swarm Intelligence.
- In computational sense, a swarm has been defined as a set of agents which are liable to communicate directly or indirectly with each other and which collectively carry out a distributed problem solving.

Current Trends in AI

- **Expert system** is a computer system that emulates the decision-making ability of a human expert. Expert systems are designed to solve complex problems by reasoning through bodies of knowledge, represented mainly as if-then rules rather than through conventional procedural code.

  For eg. DENDRAL (Dendritic Algorithm): Expert system used for chemical analysis to predict molecular structure

Current Trends in AI

- **Intelligent Agents:** It refers to an autonomous entity which acts, directing its activity towards achieving goals, upon an environment using observation through sensors and consequent actuators. Intelligent agents may also learn or use knowledge to achieve their goals.

  For eg. Alexa and Siri