ARTIFICIAL INTELLIGENCE

The Very Idea

Vasant G. Honavar
Dorothy Foehr Huck and J. Lloyd Huck Chair in Biomedical Data Sciences and Artificial Intelligence
Professor of Data Sciences, Informatics, Computer Science, Bioinformatics & Genomics and Neuroscience
Director, Artificial Intelligence Research Laboratory
Director, Center for Artificial Intelligence Foundations and Scientific Applications
Associate Director, Institute for Computational and Data Sciences
Pennsylvania State University

vhonavar@psu.edu
http://faculty.ist.psu.edu/vhonavar
http://ailab.ist.psu.edu

Turing Machine

- Hilbert and Ackerman (1928)
  - Decision Problem: "Is there an effective procedure, that allows us to decide, by means of finitely many operations, whether any given theorem is provable from a given set of premises?"
- Alan Turing (1936)
  - What is an effective procedure?
  - An effective procedure is an algorithm – step-by-step sequence of instructions that can be executed by a human or a machine
  - What kind of machine?
  - Turing Machine!

Turing's observations

Based on his observation of human computers at work, Turing concluded that a computer must be able to:

- Read the inputs of the computation (e.g., "455 + 376")
- Execute the appropriate algorithm
  - A sequence of simple steps like adding two digits in the 1's position
  - or steps in the computation which Turing called the states of mind of the (human) computer
  - which later got abbreviated as simply states
- Write results to keep track of the results of each stage.

Turing's observations

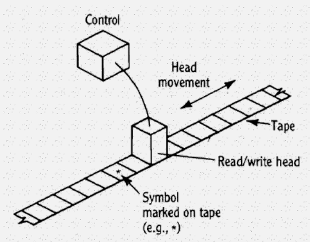
- The two-dimensional nature of the paper is irrelevant
- There is no reason to read more than one symbol at a time
- There is no reason to write more than one symbol at a time
- There is a need for the computer to step through states that keep track of the stage of the computation being carried out
  - Example: the digits in the one's position have been added and carry written

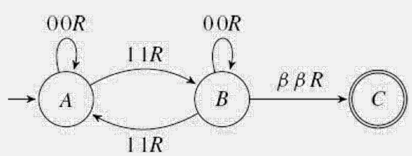
Turing's claim

- An effective procedure is simply an algorithm that tells a machine exactly what to do at any given step
- From the standpoint of executing such an algorithm human and the machine are functionally equivalent
  - Given the same input, they both can be seen as carrying out identical steps ending up with identical outputs
- The physical substrate used to implement the machine is immaterial
- All that is needed is the ability to manipulate symbols based on the syntax of the input
- One can build computers out of silicon, tinker toys, …

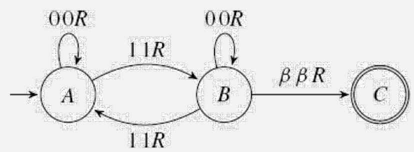
From Finite Automata to Turing Machine

- A finite automaton can be designed such that
  - The machine changes its state based on
    - Its current state and the input symbol that is read at the current position of the head
    - Writes a symbol on the tape
    - Moves the read/write head to the left or right or stays where it is





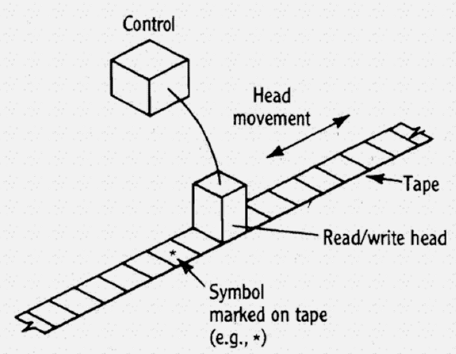
From Finite Automata to Turing Machine



- In state A, if you read a 0, write a 0, move R, and stay in A
- In state A, if you read a 1, write a 1, move R, and enter B
- In state B, if you read a 1, write a 1, move R, and enter A
- In state B, if you read a 0, write a 0, move R, and stay in B
- In state B, if you read a blank (β), write a blank, move R, and enter C

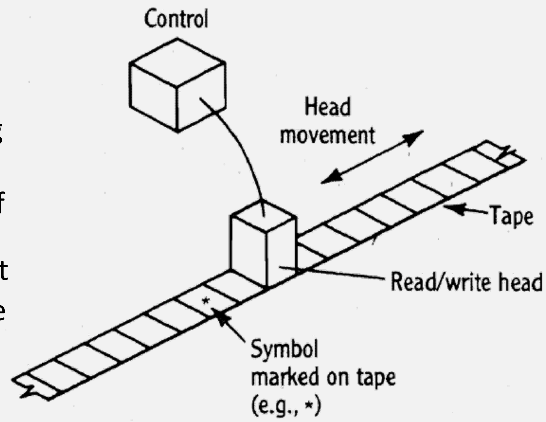
Turing Machine

- The tape is assumed to be infinite in both directions
- Ensures that there is enough space to write intermediate results
- At any given step
  - only a finite portion of the tape is used
  - the rest of the tape is blank

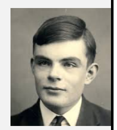


Universal Turing Machine

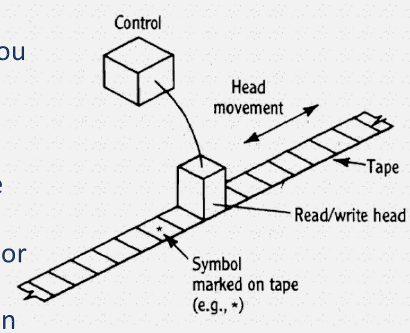
- Turing showed that there is a Universal Turing Machine that can simulate any other Turing Machine
  - Input: The program of the target Turing machine, and its input
  - Output: Output of the target machine



Turing Machine



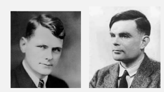
- Defines an effective procedure
  - ✓ An effective procedure is an algorithm executed by the Turing machine
- If you agree with Turing's definition, you have no choice but to accept that any algorithm whatsoever can be implemented by a Turing Machine
- If you disagree with the definition, the burden is on you
  - to devise an alternative definition or
  - at least show an example of a process that intuitively looks like an algorithm but cannot be realized by a Turing Machine



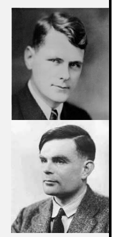
Church-Turing Thesis

- Many tried to investigate alternative formalizations of effective procedures
  - Alonzo Church proposed λ −calculus
  - Emil Post proposed Post productions
  - Andrey Markov proposed Markov Algorithms
  - Stephen Kleene proposed general recursive functions

- All of them were shown to be equivalent to Turing Machines
  - Church-Turing Thesis: Any computation that any of the formalisms could perform could be performed by the Turing Machine and vice versa



Church-Turing Thesis



- All the attempts at alternative formalizations of effective procedures were shown to be equivalent to Turing machines
  - Any computation that any of the formalisms could perform could be performed by the Turing Machine and vice versa
- Church-Turing Thesis: Any function that can be computed by an algorithm can be computed by a Turing Machine
  - What does it mean for a function to be computed by an algorithm?
    - Executing a finite sequence of instructions on a finite input producing a finite output
    - Transforming one finite sequence of letters into another sequence of letters according to a set of rules in a finite number of steps

Implications of Church-Turing Thesis

- Any computer that is sufficiently powerful (Turing equivalent) can execute any algorithm or program whatsoever
- Any program for one computer can be translated to an equivalent program for another computer
  - Why?
- Programs written in one programming language can be translated into programs written in any other programming language
  - Why?
- The design of modern computers, programming languages, software .. rests on Church-Turing Thesis

Limits of computing machines

- There exist problems that are unsolvable by Turing machines
  - Halting problem: Deciding whether an arbitrary program eventually terminates on *all* inputs
  - Theorem proving in logic: Deciding whether a theorem logically follows from a set of axioms
    - If the theorem is true, there is a finite proof
    - If the theorem is false, it is possible that the machine runs for ever
      - there is no way to tell if it is
        - because the theorem is false or
        - because the machine needs more time
  - Hilbert's decision problem is semi-decidable

Can we build computers that are more powerful than Turing machines?

- Maybe, if we can allow the machines to take advantage of something Turing machine can't
  - Infinite precision numbers
    - There is no physical substrate capable of supporting infinite precision
  - Quantum phenomena – machines that make use of quantum as opposed to deterministic states
    - There is no evidence that quantum computers can circumvent the limitations of Turing machines although they can be far more efficient than conventional computers on some problems
- We don't know if we can build computers that are more powerful than Turing Machines
- No one has succeeded to date in doing so

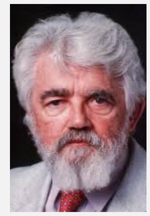
Implications of Church-Turing Thesis for AI

- So long as intelligence has an algorithmic description, there are many ways to realize it that are functionally equivalent
  - Brains
  - Digital computers
  - Quantum computers
  - DNA computers
  - And many other types of computers….

Implications of limits of computation for AI

- Position 1: To the extent that intelligence can be modeled by computation, there will be some truths that are unknowable by both humans and machines
- Position 2: Powers of mind exceed those of computers – Tantamount to saying that minds have no finite algorithmic description
- Position 3: Semi-decidability is a property of machines that reason deductively. Most of our knowledge of the world comes from inductive and not deductive methods so the limits of deductive methods are only partially relevant to AI

On Artificial Intelligence



``[This] is or should be our main scientific activity

- studying the structure of information and
- the structure of problem-solving processes
- independently of applications and
- independently of its realization in animals or humans.''

- Implicit in this is the audacious idea is a functional or naturalist view of intelligence
- McCarthy envisioned AI studying intelligence in a manner analogous to aerodynamics studies flight
  - independent of its applications and
  - independent of its realization in birds that fly

How is AI like aerodynamics?

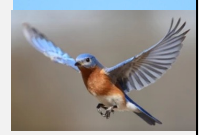
- Birds fly
- Ability to fly is natural (at least for birds)
- There are many types of bird flight (associated with different types of wings)

How is AI like aerodynamics?

- Albatrosses and some other seabirds have long, narrow wings and take advantage of winds over the oceans.

- Vultures and hawks have broad wings with slotted tips that permit more use of updrafts and winds deflected by hills.

- Birds such as ducks, pigeons, and falcons tend to have moderately long, pointed wings and fly rapidly using continuous wing beats.

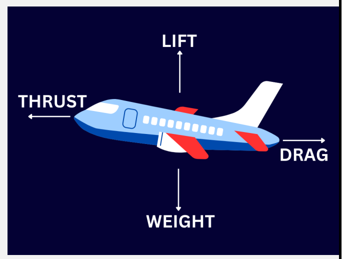


- Many songbirds use their short, rounded wings to move with quick wing beats from perch to perch or from ground to perch.

How is AI like aerodynamics?

Whether the flier is a bird or an airplane

- To become airborne, flier must exert enough lift force to counteract its weight force of gravity
    - Lift is provided by air moving over and under the wings
- To move forward, the flier must exert thrust force to overcome the drag force or air resistance
    - Muscles provide the thrust in birds
    - Engines provide the thrust in airplanes



How is AI like aerodynamics?

- Aerodynamics, the science of flight explains flight
  - How birds fly
  - How to design airplanes that fly
- Both birds and airplanes must obey aerodynamics
  - But they fly in very different ways
    - Airplanes don't flap their wings while in flight
    - The wings of airplanes are not made of feathers
  - Airplanes differ from birds more strikingly than birds differ from each other
  - What they share is the ability to fly

How is AI like aerodynamics?

- AI is about understanding the essential principles and mechanisms that underlie intelligence
  - just as aerodynamics is about understanding the essential principles and mechanisms of flight.
- Scientific understanding of intelligence should
  - Enable us to build a a variety of machines that exhibit (different aspects of) intelligent behavior
  - Help us understand minds, brains, and ultimately, ourselves, in ways that would not otherwise be possible.
- Aerodynamics : : flight : : AI: intelligence

Working hypothesis of AI and what it entails

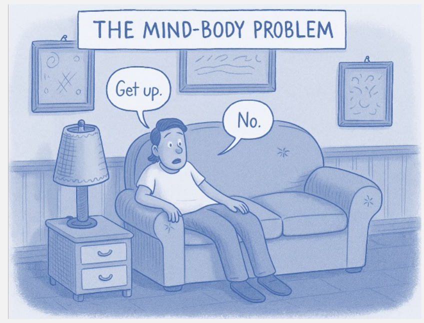
- Working hypothesis of AI
  - Cognition is (or at least can be modeled by) computation
- We will have a theory of
  - Problem-solving if we can devise algorithms that can solve problems
  - Game-playing if we can devise algorithms that play games e.g., chess
  - Reasoning if we can devise algorithms that reason from assumptions to conclusions
  - Learning if we can devise algorithms for learning from experience
  - Language if we can devise algorithms that effectively communicate using language
  - Cooperation if we have a computational model of multi-agent collaboration
  - Creativity if we can devise algorithms that exhibit creativity

Working hypothesis of AI

- Working hypothesis of AI
  - Cognition is (or at least can be modeled by) computation
- How do we know the working hypothesis of AI is true?
  - We really do not!
  - Like any other working hypothesis in science, this hypothesis is
    - subject to revision
    - or even outright replacement
    - if scientific evidence so warrants
- Yet, exploring this hypothesis has led to significant advances in
  - Our understanding of intelligence
  - Our ability to build AI systems
- Working hypothesis of AI has given birth to the computational theories of mind

A little detour: The mind-body problem

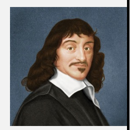
- The mind-body problem is one the oldest and the deepest questions in philosophy, neuroscience, and now even AI
- It has to do with
  - Understanding the relationship between mind (thoughts, feelings, beliefs, desires) and
  - the body (especially the brain, a physical organ)?



What is the Mind-body problem?

- Mental states
  - pain, joy, imagining a sunset, loving someone, or believing 2+2=4
  - seem to be very different from physical states like neurons firing or chemicals flowing in the brain
- Yet science shows that if there is damage to the brain, mental functions are impacted
  - Mental function appears to depend on the brain function
- The mind-body problem can be summarized as follows:
  - Are mind and brain the same?
  - Are they distinct entities?
  - If so, how are they related?

Classic positions on the mind-body problem



- Dualism (Descartes)
  - The mind is non-physical (soul, conscious) and the body is physical
  - Qualia (subjective experience) are due to non-physical mind
  - Problems
    - How can a non-physical mind run or be run by a physical physical body?
    - Scientifically untestable and somewhat magical

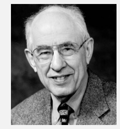
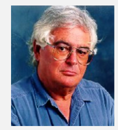
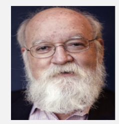
Classic positions on the mind-body problem

- Physicalism or Materialism (Smart, Place, Armstrong)
  - The mind *is* just the brain (or what the brain does)
  - Mental activity is brain activity
  - Qualia are *nothing over and above* brain processes
    - Pain is same as C-fiber firing
  - Problems
    - How do we experience the subjective feelings of pain, pleasure, being alive, being conscious, etc.?



Computational theory of mind and the mind-body problem

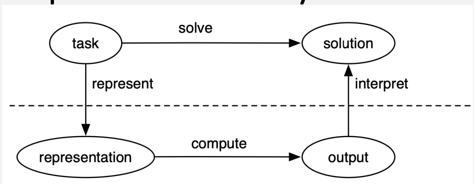
- Computational theory of mind (CTM) (Putnam, Fodor, Dennett)
  - Mind : Brain :: Software : Hardware
  - CTM a functionalist theory of mind
  - Offers a neat model of how the mind works
  - Qualia = the functional/causal role of internal states
    - what they do, not what they're made of
  - Problems
    - Is simulating a mind the same as having a mind?
    - What is consciousness? How does it arise?

Computational theory of mind and the mind-body problem

- Why did Jill get on the plane?
  - Because she wanted to visit her mother and knew the plane would take her to New York where her mother lives.
  - If she hated her mother, or if she knew the plane would take her to San Francisco instead of New York, she would not be on the plane.
- Jill's desire to visit her mother and her knowledge that the plane would fly to New York are not physical, but mental entities.
- But they have the power to trigger or cause events in the physical world, such as getting Jill to get on the plane to New York.

Computational theory of mind and the mind-body problem

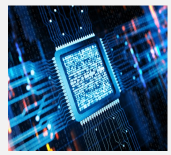
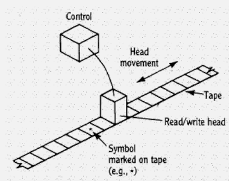


The computational theory of mind (CTM) offers a resolution of the mind-body problem.

- CTM says that contents of the mental realm, e.g., thoughts, are encoded by configurations of symbols.
- The symbols are the states of a physical entity, like the circuits s in a computer or neurons in the brain
- Symbols represent objects, events, and relationships in the physical world
  - Because they are triggered by them via our sense organs, and
  - Because of the physical effects they result in once they are triggered

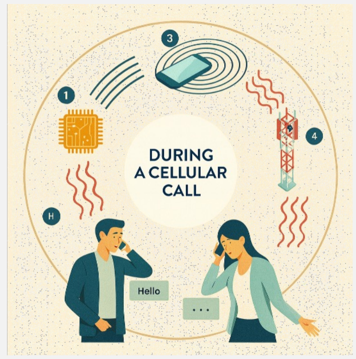
Computational theories of mind

- Cognition, or thought processes –
  - regardless of the physical substrates that realize them
    - be they living brains or silicon devices
    - can be modeled by computation

the computational theory of mind

- When you speak to your mother on your cellphone
  - A microphone turns your voice into electrical signals
  - A microchip in the phone modulates (or varies) a radio wave using the electrical signal
  - The radio wave travels through the air to a nearby cell tower
  - The tower sends your voice to your mother
  - A speaker converts the radio waves into sound so she hears your voice



Unpacking the computational theories of mind

- Your message to your mother remains unaltered as it is carried by different media
  - from vibrations of air to
  - electrons in a wire to
  - electromagnetic waves and
  - back again in the reverse order
- The same message could have been transmitted by a text or email
- Multiple physical substrates can carry the same information
- Likewise, multiple physical substrates can realize a functionally equivalent computation

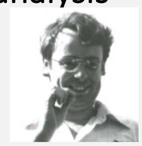
Unpacking the computational theory of minds

- Computational theory of minds implies a functional view of minds
- Minds can be modeled by computation
- A computation is realized by a program that runs on a
  - a computer made of
    - vacuum tubes
    - electromagnetic switches
    - transistors
    - integrated circuits
  - brain
  - a group of humans passing notes to each other to accomplish the same functional behavior

Unpacking the computational theory of minds

- Computational theory of minds implies a functional view of minds
  - A functional view of minds
    - Is implicit in the attempts of Hobbes, Leibniz, Boole, and Turing to explain minds in computational terms
    - Is shared explicitly or implicitly by all the work in
      - Artificial Intelligence
      - Cognitive science
      - Computational neuroscience
    - Implies that intelligence is a functional capacity independent of the specific physical substrates that support it

Computational theory of mind implies levels of analysis



- Functional view of minds implies levels of analyses

	LEVELS		
Computation	1	What	Flight
Algorithm	2	How	Aerodynamics - lift, thrust
Implementation	3	How exactly	Birds, Airplanes, Drones ..

Unpacking the computational theory of minds

- Computational theory of minds implies a functional view of minds
- Does this mean that the physical substrate –
  - whether it is the brain –
  - or the computer --
  - is irrelevant to understanding the minds?
- No!
  - Programs are executed by assemblies of simple information processing units of the sort that we saw earlier
  - Circuits made from neurons may be a lot slower than circuits made from silicon in performing complex arithmetic calculations
  - Neural circuits may be a lot faster than their silicon counterparts in making sense of a video

Computational theory of minds and AI

- Computational theory of minds implies a functional view of minds
- In this course, our focus is on
  - *what* is being computed by machines that behave as if they have minds
  - *how* it is computed by an *algorithm*
  - but not necessarily *how exactly* it is being computed
- The implementation of the computations on specific physical substrates is left to other courses
  - AI – implementation using programs written in a specific programming language executed on digital computers
  - Neuroscience – implementation in brains

Artificial Intelligence – the Very Idea

- Working hypothesis of AI
  - Cognition is (or at least can be modeled by) computation
- We will have a theory of
  - Problem-solving if we can devise algorithms that can solve problems
  - Game-playing if we can devise algorithms that play games e.g., chess
  - Reasoning if we can devise algorithms that reason from assumptions to conclusions
  - Learning if we can devise algorithms for learning from experience
  - Language if we can devise algorithms that effectively communicate using language
  - Cooperation if we have a computational model of multi-agent collaboration
  - Creativity if we can devise algorithms that exhibit creativity

Artificial Intelligence – The very idea

- John McCarthy organized a meeting of the group at Dartmouth in the summer of 1956
- "to proceed on the basis of the conjecture that every aspect of learning or any other feature of intelligence can in principle be so precisely described that a machine can be made to simulate it."

A brief (recent) history of AI

- Birth of artificial intelligence (1956)
- Early demonstrations of artificial intelligence and the publication of Computers and Thought (1959)
- 1960-1970
  - Optimism fueled by early success on some problems thought to be hard (e.g., theorem proving)
  - Slow progress on many problems thought to be easy (e.g., vision, language);
  - Fragmentation of AI into sub-areas focused on problem-solving, knowledge representation and inference, vision, planning, language processing, learning, etc.

A brief (recent) history of AI

- 1970-mid 1980s
  - Advent of practical tools such as expert systems
  - Difficulty of knowledge engineering draws attention to learning systems

A brief (recent) history of AI

- Mid 1980s-mid 1990s
  - Renewed interest in biologically inspired neural network models
  - Modest successes on some problems, e.g., vision.
  - Progress in machine learning begins to offer promising and practical alternatives to knowledge engineering.
  - Progress in the various sub-fields of AI directs attention on the design of intelligent agents that exhibit multiple aspects of intelligence

A brief (recent) history of AI

- Mid 1990s-2010
  - The advent of the World-Wide-Web and advances in computing and storage technologies make it possible for AI systems to be trained on massive amounts of data
  - Practical applications of AI in information retrieval, fault diagnosis, computer vision, information extraction from text, robotics, and related applications.
  - Major breakthroughs in learning theory offer insights that lead to practical advances, e.g., kernel machines, in machine learning.
  - Growing interest in studying multi-agent systems, including inter-agent communication, coordination, and multi-agent organizations.

A brief (recent) history of AI

- 2010-2020
  - Increased availability of data and advanced hardware, e.g., graphical processing units, spur advances in deep learning and progress on computer vision, natural language processing, and related problems.
- 2020-present
  - Large language models capture public interest and imagination
  - Advances in powerful AI systems that could automate aspects of human intellectual work raise interest in socially responsible AI for maximizing the societal benefits of AI while minimizing its potential for harm.
  - There is a subtle shift in the goal of AI from automating intelligent behavior to augmenting and extending human intellect and abilities.

Goals of AI

The long-term dream of AI is

- To build machines that have the full range of capabilities for intelligent action that people have
- To build machines that are self-aware, conscious, and autonomous in the same way that people like you and I are.''

Michael Wooldridge in "A Brief History of Artificial Intelligence"

- This is the science fiction vision of AI you see in movies
- We don't really understand what such an effort entails - recall the various tests of intelligence
- There is little consensus on whether human-like AI is feasible, or even desirable.

Scientific goal of AI

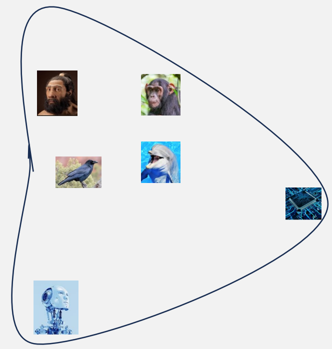
The central scientific goal of AI is to understand intelligence by building and studying computational models of intelligence.

- We will have a theory of
  - Problem-solving if we can devise algorithms that can solve problems
  - Game-playing if we can devise algorithms that play games e.g., chess
  - Reasoning if we can devise algorithms that reason from assumptions to conclusions
  - Learning if we can devise algorithms for learning from experience
  - Language if we can devise algorithms that effectively communicate using language
  - Cooperation if we have a computational model of multi-agent collaboration
  - Creativity if we can devise algorithms that exhibit creativity

Scientific goal of AI

AI is the science of exploration of the space of possible and actual intelligent systems.

- The intelligence of humans and animals provide existence proofs or examples of designs for intelligent systems found in nature.
- Exploring this design space entails characterizing the existing designs and conceiving and evaluating alternative designs with the desired characteristics and performance.



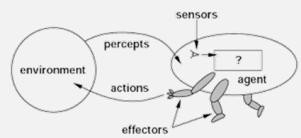
Practical goal of AI

- AI is is the enterprise of the automating tasks that are believed to require intelligence when performed by humans
  - proving theorems
  - planning trips
  - recognizing faces
  - diagnosing diseases
  - designing computers
  - composing music
  - discovering scientific laws
  - proving mathematical theorems
  - playing chess, writing stories
  - teaching physics
  - negotiating contracts
  - providing legal advice

Practical goal of AI

- AI is about augmenting and extending human intelligence, problem solving abilities, and creativity.
  - An AI physician's assistant helps medical practitioners make better decisions
  - A search engine augments human memory
  - Natural language translation systems help people communicate across linguistic barriers
  - An AI writer’s assistant can act as a muse
  - AI-powered scientist's assistants help identify promising hypotheses to pursue, optimal experiments to run, and help analyze and interpret experimental results

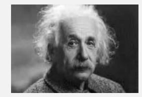
Practical goal of AI



- AI is about the analysis and synthesis of agents that exhibit intelligent behavior.
- An agent is an entity that that senses or perceives the environment acts on the environment.
- The agent is considered intelligent
  - to the extent that its actions are appropriate given its circumstances and perceptual and computational limitations
  - and its behavior is adaptive to changes in its circumstances and in the environment
  - and its behavior improves with experience (through learning).

Success measures

Not everything that can be counted counts and not everything that counts can be counted



- Success measures depend on the goals
  - Successful automation of some aspect of intelligent behavior simply requires that the AI system perform the tasks at hand with a level of competence that makes it useful in practical settings.
  - Effectively augmenting or extending the capabilities of a physician in an emergency room calls for complementing and not duplicating what the physician is good at so that the human-AI team achieves outcomes that are superior to what either could on its own.
- Choosing appropriate measures is critical to progress in AI

Relationship pf AI to other disciplines

AI in relation to computer science

- AI has a special relationship to computer science because computation : mind :: calculus : physics
- The relationship of AI to Computer Science is like the relationship of physics to mathematics
- Mathematics offers essential tools for physics, but physics is not just mathematics
- The objects of study in mathematics can be entirely abstract with no relationship to the physical world or experimental measurements
- When mathematics is employed in physics, the definitions are abstractions or idealizations of entities in the natural world.
- Consequently, physical theories have to be verified through experiments

AI in relation to computer science

- AI takes advantage of advances in computer science
- AI contributes concepts and tools for computer science
- AI stimulates advances in computer science
- AI is not a subset of computer science.
- Computer science is not a subset of AI

AI in relation to psychology

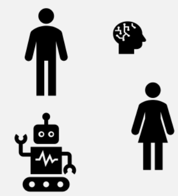
- AI is often seen as a sibling of psychology
- Psychology is concerned with studies of human and animal behavior
- AI is concerned with computational models or artifacts that exhibit intelligent behavior
- AI is not committed to human-like mechanisms or any particular implementation of such mechanisms
- Computational models from AI have influenced contemporary research in psychology
- Findings and insights from psychology often have informed the design of AI models

AI in relation to neuroscience

- AI has deep connections to neuroscience
- Neuroscience is about brains
- AI is about computational theories of brain funcion
- AI work continues to be informed by or at least inspired by findings in neuroscience
- Advances in machine learning offer new perspectives on old questions in neuroscience

AI in relation to social sciences

- Insofar as AI is concerned with the design of intelligent agents that
  - act rationally
  - interact with other agents
- AI draws on ideas from economics, game theory, organizational theory, decision theory, and other areas of social sciences.
- AI systems can inform studies that drive advances in the social sciences.



AI in relation to engineering

- Insofar as AI is concerned with the design of intelligent artifacts, it both contributes to, and draws on advances in engineering.
- AI advances have resulted in practical tools for
  - configuring computer systems
  - Diagnosing faults in machinery
  - software agents that scour the Internet for information on demand
  - Intelligent systems for planning and scheduling
  - Computer-aided design tools in many engineering disciplines
  - Self-driving automobiles, Smart buildings, smart robots, etc.

Some lessons from AI – or why is it hard to realize AI?

Easy problems for AI
- Arithmetic, algebra, logic
- We have precise algorithms to instruct computers

Somewhat hard problems for AI
- Board games like Chess, Backgammon, etc.
- Effective play requires looking ahead many moves – search space too large – need heuristics

Moderately hard problems for AI
- Vision, language translation, etc.
- No known algorithm – train machine learning on large data

Even harder problems for AI
- Creativity

Hardest problem for AI
- Artificial General Intelligence

AI and society

- AI will disrupt all areas of life
  - Automated driving can reduce accidents and save lives
  - Automated driving will result in significant job loss
  - Similar dilemmas are presented by other areas – accounting, healthcare, journalism, banking
  - What will happen to the workers who once occupied those jobs?
  - Will new jobs be created?
  - How can workers get trained for the new jobs?
  - How can we design systems and prepare society to best leverage the complementary strengths of humans and AI?
  - How can we ensure that AI systems do not become instruments of injustice, human rights violations, and oppression?
  - How can we maximize the benefits of AI while minimizing its potential for harm