Philosophy and AI
Lecture 6: Philosophy of Mind II

Marco Degano

19 February 2025

Readings

Required:

- Glymour 2015: chapter 14, pp. 347 – 349; 351 – 365
    - Required: Intro; The Computational Conception of Mind; Are Mental States in the Head?; The Chinese Room; Challenges to the Computational Conception of Mind; Can the Computational Conception of Mind Be Wrong?
    - Optional: Bounded Rationality; Rationality and Computationally Bounded Systems; Conclusion.

Optional:

- Rescorla, Michael, "The Computational Theory of Mind", The Stanford Encyclopedia of Philosophy, `https://plato.stanford.edu/archives/win2024/entries/computational-mind/`
- *Minds and Machines* MIT OCW, Part 1, Minds and Computers `https://openlearninglibrary.mit.edu/courses/course-v1:MITx+24.09x+3T2019/course/`

Outline

Outline

Introduction

- The computational theory of the mind (CTM) explores how mental processes can be modeled as computations, forming a foundational idea in philosophy and AI.
- This idea has deep historical roots, from Carnap's construction of the world from sense data to modern AI's use of algorithms to model cognition.
- We will:
  - Introduce the computational theory of mind.
  - Examine key arguments for and against it.
  - Explore connections to current AI systems and alternative perspectives.

The computational theory of mind

- **Functionalism:** Mental states are characterized by what they do-not by the material they are made of.
- **Computational theory of mind**: mental states are computations, meaning the mind functions like a computer, processing information through formal rules, algorithms, or symbolic representations
- The mind's behavior can be described as performing computations, irrespective of how it is realized physically.
- This computational perspective allows us to abstract complex mental processes into manageable, describable systems.

The computational theory of mind

Historical Development:

- **17th century:** Hobbes viewed reasoning as "computing" with symbols.
- **19th century:** Neuropsychologists proposed the brain as a machine, with nerve cells acting as computational units.
- **20th century:** Turing, Gödel, Church, and others formalized theories of computation. Turing's idea of machines capable of symbolic manipulation laid a foundation for AI.
- **Present:** Cognitive science and AI describe the mind using models such as neural networks and reinforcement learning, reflecting how our brain processes information.

The computational theory of mind

Two perspectives on the computational mind:

- **Literal Computation:** The mind is akin to a Turing machine-its mental states are actual computational states. This view suggests that by understanding and replicating the formal rules, we can fully explain cognition.
- **Computational Metaphor:** Alternatively, computation serves as a powerful metaphor or abstraction for understanding complex mental processes. Neural networks, for instance, do not strictly follow symbolic rules but approximate cognitive functions.

Marr's Three Levels of Analysis

- David Marr proposed that cognitive systems should be understood at three complementary levels:
  - Computational Level: **What is the problem?** What does the system do, and why?
    Example: A facial recognition system solves the problem of identifying faces in images by analyzing patterns.
  - Algorithmic Level: **How is the problem solved?** What representations and processes (algorithms) are used to transform input into output?
    Example: It uses convolutional neural networks (CNNs) to extract features and classify images.
  - Implementational Level: **How is it physically realized?** What is the physical substrate that carries out the algorithm?
    Example: The CNN operates on GPU hardware designed for matrix computations.

More Examples

Vision:

- **Computational:** Determine 3D shape from 2D retinal images.
- **Algorithmic:** Extract edges, compute depth cues, and build a 2.5D sketch of the scene.
- **Implementational:** Neural circuits in the retina and visual cortex that process visual signals.

Language Processing:

- **Computational:** Understand and generate meaningful sentences.
- **Algorithmic:** Use models (e.g., transformer architectures) to capture syntactic structure and semantic relationships.
- **Implementational:** Run on modern hardware (CPUs/GPUs) and optimized neural network libraries.

More Examples

A well-known example: **flight** in **birds** vs **airplanes**

- **Computational Level:**
  - *Problem:* Achieve and sustain flight by generating lift and controlling movement.
- **Algorithmic Level:**
  - *Birds:* Use flapping wings, flexible wing shape, and dynamic adjustments to modulate airflow.
  - *Airplanes:* Rely on fixed-wing aerodynamics, with engines providing thrust and control surfaces for stability and steering.
- **Implementational Level:**
  - *Birds:* Biological systems with lightweight skeletons, muscles, and feathers.
  - *Airplanes:* Engineered structures built from metals and composites.

Levels and AI

Key Insight: These levels highlight different aspects of cognition and AI. For instance:

- Evaluating an AI system like GPT involves studying its training objective (computational), its transformer architecture (algorithmic), and its hardware acceleration (implementational).

Relevance: So when talking about 'the behavior of the mind' its good to distinguish at which level we're talking about it.

Outline

Overview

- The Chinese Room Argument, introduced by John Searle, challenges the claim that running a computer program is sufficient to produce genuine understanding.
- It highlights that while computers may simulate intelligent behavior through formal symbol manipulation (i.e., syntax), this process is not enough to generate semantic understanding or true mental states.

The Thought Experiment

- **Scenario:** Imagine a monolingual English speaker confined in a room with:
  - A large collection of Chinese symbols.
  - An English instruction manual detailing how to manipulate these symbols.
- **Input/Output:** The person receives Chinese characters as input and, by following the rules, produces appropriate Chinese responses.

The Thought Experiment

- **Illusion of Understanding:** To external observers, the responses appear indistinguishable from those of a native Chinese speaker.
- **Key Insight:** Despite producing coherent answers, the person does not understand Chinese-they are merely executing syntactic rules without grasping any meaning.

Implications for AI

- Simulation vs. Genuine Understanding: Searle concludes that merely simulating understanding (i.e., producing correct responses) is not equivalent to truly having understanding.
- **Impact on AI:**
  - Raises doubts about whether AI systems (at least those operating on symbol manipulation) can possess genuine semantic understanding.
  - Suggests that additional factors-possibly linked to the biological nature of human brains-may be necessary to achieve true consciousness.
- **Broader Philosophical Debate:** Challenges computational theories of mind and urges us to reconsider what it means for a system to “understand”.

Criticisms and Alternative Perspectives

- **Systems Reply:** Critics argue that while the individual inside the room does not understand Chinese, the system as a whole might. Searle rebuts this by suggesting that even if the person internalized all the rules, understanding would still not emerge.
- **Robot Reply:** Some propose embedding the system in a robot with sensory inputs so that symbols are grounded in real-world experiences. Searle maintains that additional inputs alone cannot bridge the gap from syntax to semantics.
- **Emergence from Complexity:** Others believe that genuine understanding may emerge from sufficiently complex computations (emergent system).
- **Takeaway:** The Chinese Room remains a provocative thought experiment, essential for exploring the distinctions between simulating intelligence and actually possessing a mind.

Outline

Overview of Challenges

- The Chinese Room Argument (and related symbol grounding concerns) question whether syntactic manipulation can yield genuine understanding.
- The Frame Problem challenges how a computational agent filters relevant from irrelevant information in a dynamic world.
- The difficulty of finding the right program-that is, selecting or designing a computational model that both replicates observed behavior and faithfully mirrors internal cognitive processes.
- Reconciling bounded rationality (our real-world cognitive limits) with the idealized norms of Bayesian or logical reasoning.

Challenge 1: The Frame Problem

- **Issue:** How can a computational agent predict the relevant consequences of its actions while ignoring the vast array of irrelevant possibilities?
  - **Example:** When flipping a light switch, an agent should consider the presence of electricity but need not factor in highly improbable events (e.g., alien invasions).
- **Current Approaches:**
  - Symbolic AI employs non-monotonic logics (e.g., as in PROLOG) to focus on typical scenarios and handle exceptions.
  - Subsymbolic AI (e.g., neural networks) learns relevance implicitly from data, though these heuristics often lack transparency and are difficult to verify.

Challenge 2: Finding the Right Program

- **Issue:** Many computational models can replicate observed behavior, but which one truly captures the inner workings of cognition?
    - **Input Problem:** How does an agent convert raw sensory data into symbolic representations suitable for computation?
    - **Underdetermination:** The same behavior may be explained by multiple, inequivalent programs.
- **Empirical Avenues:**
    - Introspection: Analyzing human-reported reasoning steps.
    - Neuroscience: Using tools like fMRI and EEG to correlate brain activity with computational processes.

Challenge 3: Bounded Rationality vs. Normative Standards

- **Issue:** Normative models (e.g., Bayesian probability, classical logic) prescribe how we *should* reason. Yet human reasoning is constrained by time, information, and cognitive resources.
  - Cognitive biases and heuristics (as shown in Kahneman & Tversky's work) illustrate these limitations.
- **Key Questions:**
  - Can computational models approximate rationality without assuming idealized processing?
  - Might alternative logics (e.g., non-monotonic logics) better capture human reasoning under constraints?
- **AI Relevance:** Systems like AlphaGo optimize decision-making within bounded resources, but CTM must explain why human reasoning deviates from these ideal norms.

Additional Considerations

- **The Symbol Grounding Problem:** How do abstract symbols acquire intrinsic meaning rather than merely having "derived" meaning from external interpretation?
- **The Explanatory Gap:** Even if we can model input-output behavior computationally, how do we account for the qualitative, subjective aspect of experience?
- Such issues drive research into hybrid models (e.g., connectionism, embodied cognition) that may better capture the richness of human mental processes.

Outline

Objection to CTM: Discrete vs. Continuous

- **Objection to CTM:** The traditional Computational Theory of Mind (CTM) models mental processes as discrete, symbolic computations (e.g., Turing machines).
- The question: Is this framework adequate to describe the computation that occurs in the brain?

Key Insight: The Brain as a Continuous Dynamical System

- **Key Insight:** The brain is not a discrete machine like a Turing machine; instead, it operates as a continuous dynamical system.
- Neurons exhibit continuous, graded activations.
- Neural signals interact continuously, constantly updating and influencing each other.

Analogy to Natural Dynamical Systems

- The brain's continuous operation resembles other dynamical systems, such as:
  - *Planetary motion*: smooth, continuous trajectories governed by gravitational forces.
  - *Flowing rivers*: exhibiting non-discrete, continuous changes over time.
  - *Chemical reactions*: continuous concentration changes and reaction rates.
  - *Thermostats*: which adjust temperature gradually based on feedback loops.

Analog Computation in Dynamical Systems

- Dynamical systems can perform computation in an analog fashion:
  - **Input:** The system's initial state.
  - **Program:** The continuous dynamics (e.g., differential equations) that govern state evolution.
  - **Output:** The evolved state after a certain period of time.
- This is in contrast to the step-by-step, discrete operations of a Turing machine.

Neural Networks and Subsymbolic Computation

- Neural networks in AI are inspired by continuous neural dynamics.
- They compute through subsymbolic processes, relying on continuous activation patterns instead of rigid symbolic rules.
- This approach aligns with how the brain operates as a dynamical system.

Embodied Cognition and Feedback Loops

- **Embodied Cognition:** The brain is not isolated in processing information but works as part of a **feedback loop** between the brain, body, and environment.
- Cognition arises from this ongoing interaction, influencing and being influenced by sensory input, motor actions, and environmental context.
- CTM provides a formal framework for cognition, it may not fully capture the continuous, dynamic, and embodied nature of human thought.

Exercises

1. Recall the twin earth argument: What does it say when applied to the computational concept of mind? (One view: the internal component of a mental state is the computational state of the computational system describing the mind, and the external component of the mental states describes what the computational representations refer to in the world.)
2. Given the objection of the embodied cognition approach, do you think it is still worthwhile to describe the abilities of the mind in terms of symbolic computation? Are Marr's level helpful for your reply?