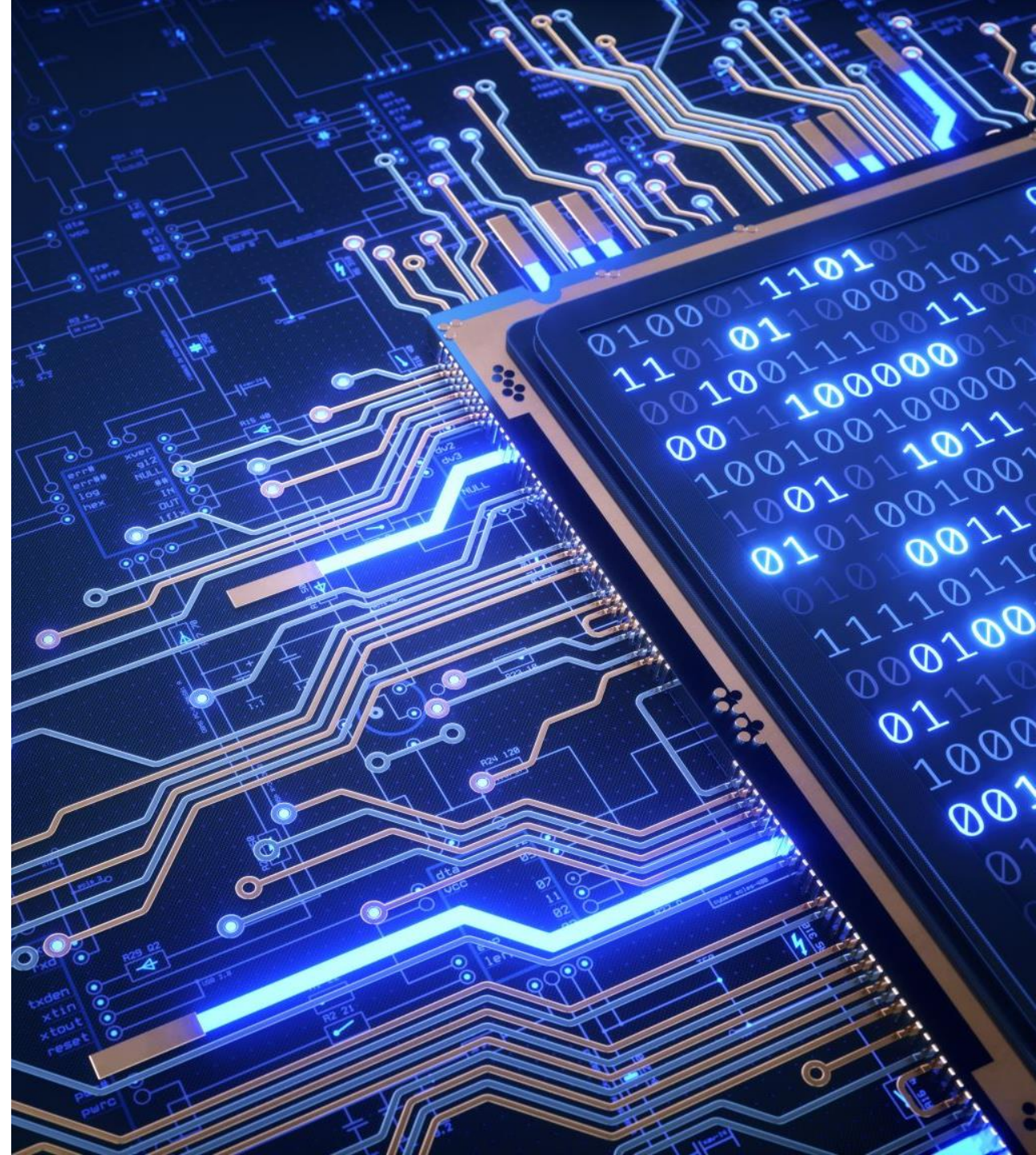


Modern computation theory and some distinctions in the metaphysics of mind

PHIL351

PLAN

- Housekeeping
- Conclude discussion of the behavioural markers of intelligence
- Turing machines and the origins of modern computation
- Theoretical significance of Turing computation for cognitive science





Housekeeping

- instructions for first paper now posted – questions?
- Stay tuned for TA office hour announcements

Review

- Issues with the Turing test *as* a test
- How else might we recognize intelligence in nonhumans, if not success at the Turing test?
- “Intelligence is a fixed goal with variable means of achieving it” (William James)
 - Appropriately goal-directed behaviour that is also flexible, transferable, and manipulable. (Fridland)
 - Appropriately goal-directed behaviour is sufficient to at least *appear* intelligent (cf. the Sphex wasp).



Flexibility/response-freedom

- The (mythical) Sphex illustrates that appropriately goal-directed behaviour is not intelligent unless it is also *flexible*; i.e., unless the system's responses are just one among many possible alternatives that the creature could have performed ('response freedom'), rather than mandatory given the stimulus.
- The converse might also seem true: flexible responses, as such, are not intelligent unless also appropriately goal-directed (e.g., purely random responses to stimuli would be 'flexible' but not intelligent.)

Transferability

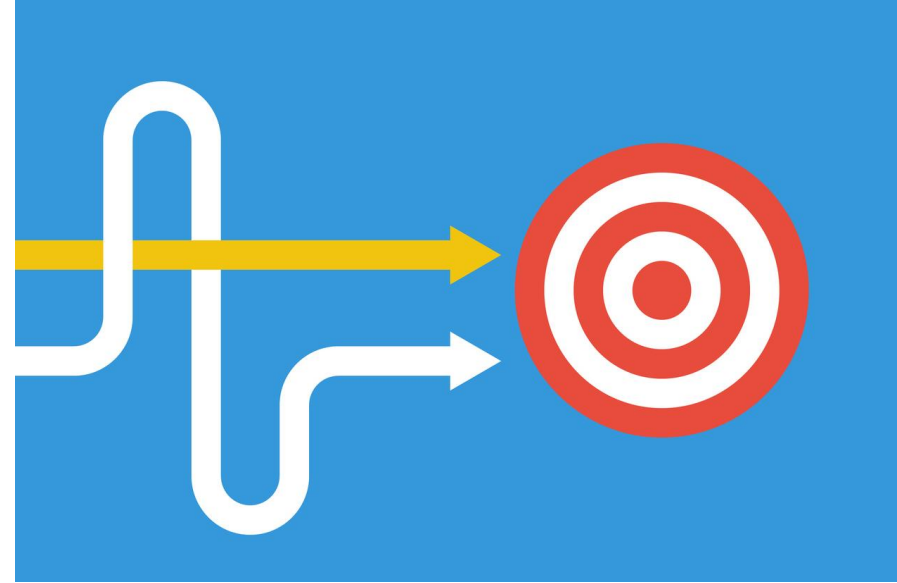
- If flexibility is the capacity to execute many different performances in the same situation, transferability is the capacity to execute the same type of performance in many different situations.
- If flexibility concerns the ability to appropriately vary one's responses, transferability concerns the ability to redeploy a certain capacity in novel ways across a wide range of counterfactual situations.
 - As with flexibility, what matters for intelligence is not mere transfer, but *appropriate* transfer.

Manipulability/endogenous control

- According to many, a paradigmatically intelligent behaviour is endogenously controlled by the agent (in light of their goals), rather than exogenously by the stimulus.
 - ‘top-down’ vs. ‘bottom-up’
 - ‘goal-directed’ vs. ‘stimulus-driven’
 - ‘active’ rather than ‘passive’

Putting these features together

- An intelligent system, in contrast with a nonintelligent one, is one that tends to arrive at the same goal state within a wide range of counterfactual scenarios of environmental variance (e.g., initial state and intermediate states).
 - Goal attainment is ‘robust’ or ‘resilient’ to perturbations.



What makes language use so intelligent?

- With these criteria, we can see why so many have seen verbal behaviour as a gold standard for intelligence.
- Having a conversation is a goal-directed activity that is subject to various norms of appropriateness. When successful, it manifests significant flexibility and context-sensitive transferability. Accordingly, the result is often highly creative.
 - Cf. Descartes on what seemed missing in the behaviours of mere brutes.
 - Cf. Lady Lovelace's objection to Turing that no mere machine would ever *innovate* or do something *novel*.

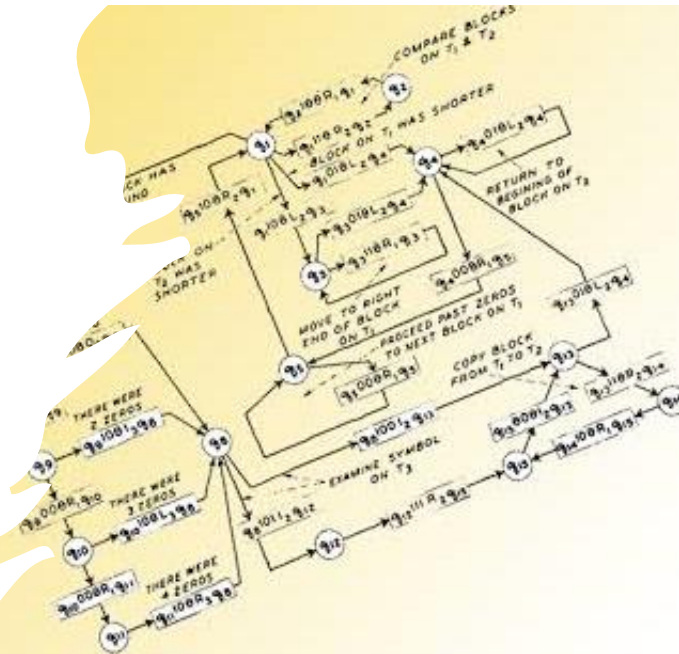


What makes verbal behaviour so intelligent?

- The same general criteria allow us to see human linguistic activity as just one among many possible behavioural expressions of intelligence. Specifically, they allow us to demarcate various kinds of intelligence and scales of intelligence.
- Showing versus telling:
<https://www.youtube.com/watch?v=cbSu2PXOTOc>



Turing machines and computation



‘Turing machines’

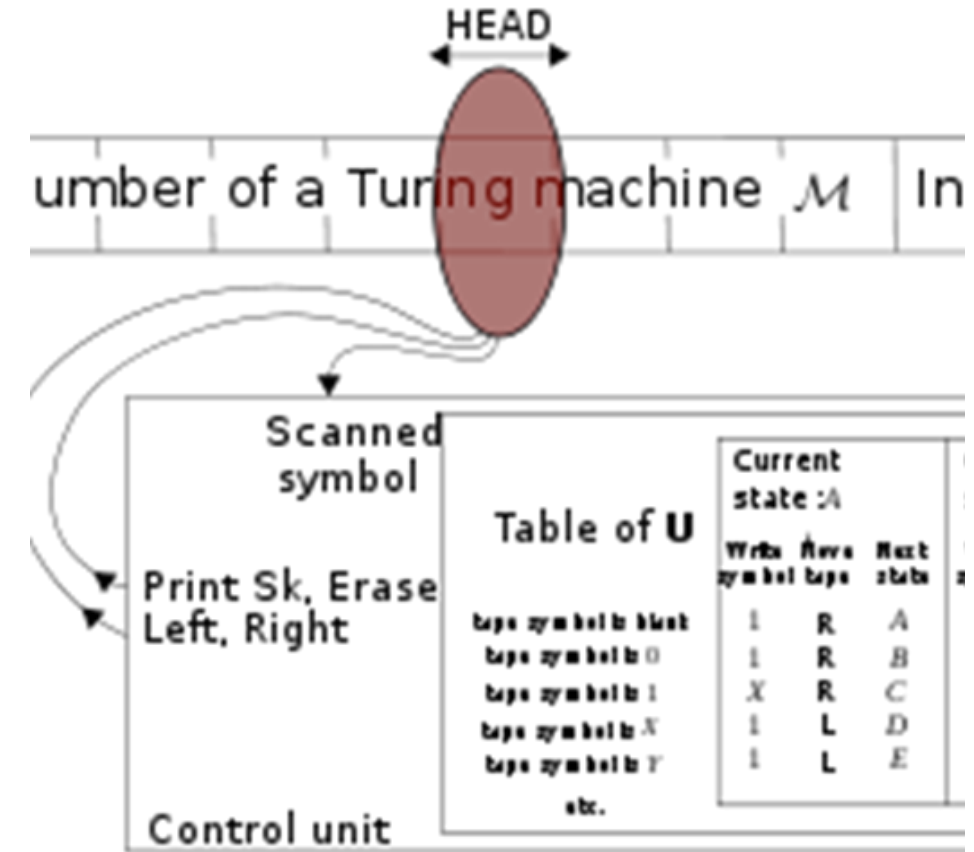
A Turing machine is a way of formalizing the notion of an algorithm or effective method: a kind of fool-proof *recipe* for achieving a certain result.



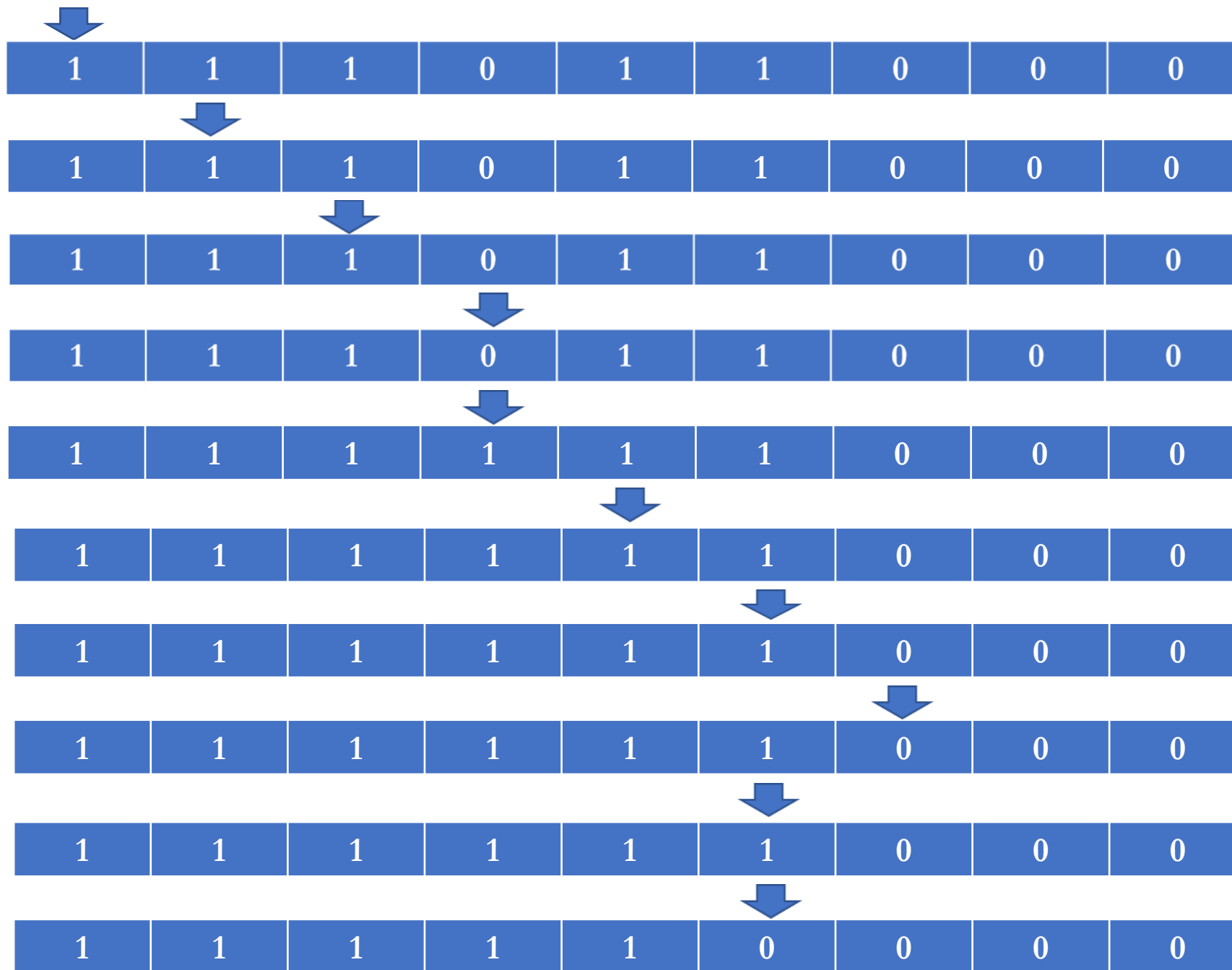
Historical background for Turing machines

- In Turing's time, mathematicians had become self-conscious about their reliance on intuitions about what is self-evidently or certain (e.g., about what should be treated as axiomatic).
 - Attempts to derive Euclid's 5th postulate resulted in a new geometry;
 - Attempts to derive arithmetic from naive set theory resulted in paradox
- Led to the 'formalism' or 'Hilbert's program' in (meta)mathematics: Mathematics is *not* the intellectual intuition of eternal platonic truths, but a 'formal system' of *meaningless symbol tokens* manipulable according to rules that are purely *stipulated* and *mechanical* (i.e., algorithmic).
 - Idea: side-step any appeal to human intuition.

A 'Turing machine' is an ideal machine composed of an infinitely long piece of tape that is divided into cells. In each cell, a symbol can be written and erased. There is also a 'machine head' that, at any given time, is located over one of the cells on the tape and that can execute specific operations: it can *read* the symbol written on the cell, *erase* the symbol in the cell, *write* a new symbol in the cell, and *move* one cell to the left or right. Which operation it executes at a given moment is a function of its 'machine table' and of the machine's internal state (of which there are finitely many). The machine table is the set of instructions that govern how the machine head behaves as a function of its internal state and the symbol inscribed in a particular cell.



Significantly, this leaves no room for ambiguity and requires no appeal to thought or judgment in the specification of the rules to be followed (exactly as required by the notion of an algorithm). It does this by being entirely *syntactically* governed.





‘Universal’ Turing Machines

Having demonstrated the existence of a Turing machine for every (Turing computable) function, Turing proved the existence of a ‘Universal Turing machine:’ a machine that can take the machine table of any special Turing machine (like the one we just described) and then run that machine.

This is the core idea behind a general-purpose – i.e., programmable – digital computer with which we are all familiar today.

Implications for cognitive science

Note what this machine just did: it successfully executed a *semantic* task: it calculated (in the language of binary) that the sum of 3 and 2 is 5. And yet, at no point did we require the machine to understand or have insight into what the symbols *mean* or to grasp an inference rule when transitioning its state. At each stage, its operations were sensitive to only *purely syntactic* properties of the symbols (e.g., the shapes of the numerals).

Hence, a paradigmatically *rational* process (here, calculation) could be executed purely *mechanically*. And the same will be true for any ‘recursive’ or ‘Turing computable’ operation.

- Descartes’ paradox of mechanical reason solved?

Implications for cognitive science: The escape from behaviourism

At the time Turing was writing, behaviourism was prominent in psychology. The behaviourists operated with a blanket prohibition of cognitive (e.g., symbolic) processes causally mediating sensory input and motor output. In their view, to appeal to such processes as part of a mechanistic explanation of behaviour would be to move in a vicious circle (e.g., to smuggle in a tiny 'homunculus' who is interpreting and manipulating symbols in light of their meaning).

With Turing machines, psychologists were in a position to lift this restriction.