The Measure of Intelligence

FRANÇOIS CHOLLET

Paper Review by Hongzheng Wang

Basic Idea

- The current AI measuring is limited.
  - The mainstream AI community tend to define and measure intelligence as the skill for specific tasks, such as board games and video games.
  - The skill is heavily modulated by prior knowledge and experience.
- This paper articulate a new formal definition of intelligence
  - The intelligence is defined as skill-acquisition efficiency.
  - Focus on the concepts of scope, generalization difficulty, priors, and experience.
- Present a new benchmark for general intelligence comparisons between AI systems and humans.

Context and history

- The intelligence needs explicit and actionable definition and measure.
  - There is an explicit promise about AI from the 1950s, which is to develop machines that possess intelligence comparable to that of humans. But AI has since been falling short of its ideal.
- If the only successes of AI have been in developing narrow, task-specific systems, it is perhaps because only within a very narrow and grounded context we have been able to define our goal sufficiently precisely, and to measure progress in an actionable way.
- Little attention to the definition or evaluation of general intelligence.

Context and history – two divergent visions

- Intelligence measures an agent's ability to achieve goals in a wide range of environments.
  - Summary of no few than 70 definitions from the literature.
  - Two characterizations: task-specific skill ("achieving goals"), generality and adaptation ("in a wide range of environments").
- Intelligence as a collection of task-specific skills.
  - Set of static programs.
  - Marvin Minsky: "AI is the science of making machines capable of performing tasks that would require intelligence if done by humans"
- Intelligence as a general learning ability.
  - Machine learning.
  - John McCarthy (paraphrased by Hernandez-Orallo): "AI is the science and engineering of making machines do tasks they have never seen and have not been prepared for beforehand"

Context and history – AI evaluation

- Skill-based, narrow AI evaluation
  - Human view: Turing test and its variants.
  - White-box analysis: mathematical proof or puzzle game.
  - Peer confrontation: compete against either other AIs or humans, like chess.
  - Benchmarks: using “test set” and score the response, like Kaggle competitions.
- Problems:
  - AI effect
  - Kaggle models are often overly specialized
  - No conditions on how the system arrives at this performance
- Intelligence lies in the process of acquiring skills.

Context and history – AI evaluation

- Generalization: robustness, flexibility, generality
  - the ability to handle situations (or tasks) that differ from previously encountered situations.
- System-centric generalization
  - New to the system, e.g., training set and test set.
- Developer-aware generalization
  - New to the system and developer, e.g., test data outside the “development set”

Context and history – AI evaluation

- Degrees of generalization for information-processing systems
  - Absence of generalization
    - such as exhaustive search algorithm
  - Local generalization, or “robustness”
    - adaptation to known unknowns within a single task or well-defined set of tasks
    - current machine learning
  - Broad generalization, or “flexibility”
    - adaptation to unknown unknowns across a broad category of related tasks
    - a domestic robot capable of passing Wozniak’s coffee cup test
  - Extreme generalization
    - adaptation to unknown unknowns across an unknown range of tasks and domains
    - human intelligence, g-factor

Context and history – AI evaluation

- The psychometrics perspective (IQ test)
  - Evaluate broad cognitive abilities as opposed to task-specific skills.
  - Ability is an abstract construct; skill is directly measurable.
  - Psychometrics approaches the quantification of abilities by using broad batteries of test tasks rather than any single task, and by analyzing test results via probabilistic models.
    - The tasks should be previously unknown to the test-taker.
  - Multi-task benchmarks in AI field: such as SuperGLUE.
    - Problem: the set of tasks is still known in advance to the developers of any test-taker, do not directly assess flexibility or generality.

Context and history – AI evaluation

- Principles of psychometrics can inform intelligence evaluation in AI in the context of the development of broad AI and general AI:
  - Measuring abilities (representative of broad generalization and skill-acquisition efficiency), not skills.
  - Doing so via batteries of tasks rather than any single task, that should be previously unknown to both the test taking system and the system developers.
  - Having explicit standards regarding reliability, validity, standardization, and freedom from bias.

A New Perspective

- The hallmark of broad abilities is the power to adapt to change, acquire skills, and solve previously unseen problems – not skill itself.

- Research on developing broad in AI systems (up to "general" AI) should focus on defining, measuring, and developing a specifically human-like form of intelligence, and should benchmark progress specifically against human intelligence.
  - characterizing and measuring intelligence is a process that must be tied to a well-defined scope of application, and at this time, the space of human-relevant tasks is the only scope that we can meaningfully approach and assess.
- An actionable test of human-like general intelligence should be founded on innate human knowledge priors.

Human Knowledge Priors

- Low-level priors about the structure of our own sensorimotor space.
  - e.g., reflexes such as the vestibulo-ocular reflex, the palmar grasp reflex, etc.
- Meta-learning priors governing our learning strategies and capabilities for knowledge acquisition.
  - the assumption that information in the universe follows a modular-hierarchical structure
- High-level knowledge priors regarding objects and phenomena in our external environment.
  - priors about orientation and navigation in 2D and 3D Euclidean spaces, goal-directedness

Human Core Knowledge

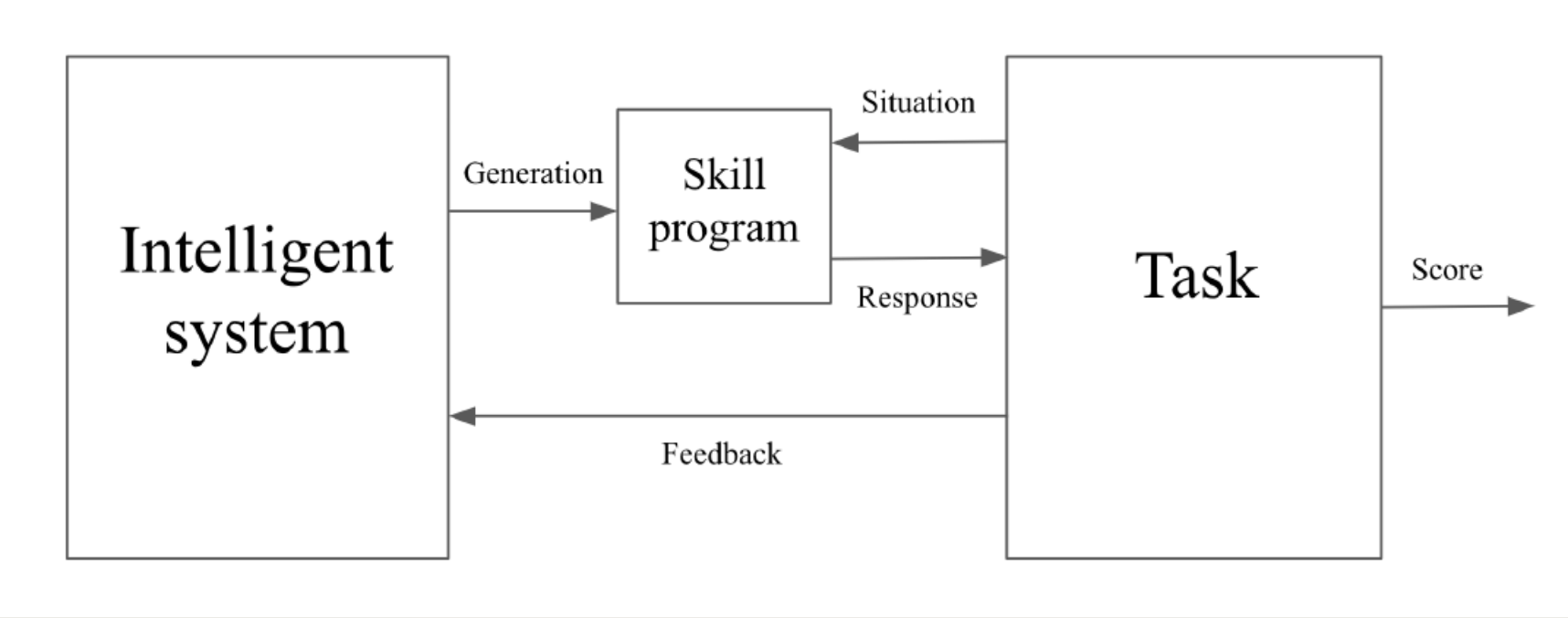
- Objectness and elementary physics
  - humans assume that their environment should be parsed into “objects” characterized by principles of cohesion, persistence, and contact.
- Agentness and goal-directedness
  - possessing intentions, acting to achieve goals
- Natural numbers and elementary arithmetic
  - number representations may be added or subtracted, compared to each other, or sorted.
- Elementary geometry and topology
  - distance, orientation, in/out relationships for objects in our environment and for ourselves

Definition of Intelligence

- The intelligence of a system is a measure of its skill-acquisition efficiency over a scope of tasks, with respect to priors, experience, and generalization difficulty.
  - Intelligence lies in broad or general-purpose abilities; it is marked by flexibility and adaptability (i.e., skill-acquisition and generalization), rather than skill itself.
  - A measure of intelligence should imperatively control for experience and priors.
  - Intelligence and its measure are inherently tied to a scope of application.
- We do not purport that the definition above and the formalism below represent the “one true” definition.

Position of the Problem

- An intelligent system generates a skill program to interact with a task



Position of the Problem

- A task T
  - A task state $TaskState$ (binary string).
  - A "situation generation" function $SituationGen: TaskState \rightarrow Situation$. It may be stochastic.
  - A "scoring function" $Scoring: [Situation; Response; TaskState] \rightarrow [Score; Feedback]$. It may be stochastic.
  - A self-update function $TaskUpdate: [Response; TaskState] \rightarrow TaskState,$ which mutates the task state based on the response to the latest situation. It may be stochastic.

Position of the Problem

- An intelligent system IS
  - A system state $ISState$ (binary string).
  - A "skill program generation function":
    $SkillProgramGen: ISState \rightarrow [SkillProgram; SPState]$. It may be stochastic.
  - A self-update function
    $ISUpdate: [Situation; Response; Feedback; ISState] \rightarrow ISState,$ which mutates the system's state based on the latest situation and corresponding feedback. It may be stochastic.

Position of the Problem

- The interaction between task, intelligent system, and skill programs is structured in two phases: a training phase and an evaluation phase.
  - The goal of the training phase is for the *IS* to generate a high-skill skill program that will generalize to future evaluation situations.
  - The goal of the evaluation phase is to assess the capability of this skill program to handle new situations.

- Also, based on the problem setup, we can define a list of useful concepts.
  - Curriculum: Sequence of interactions (situations, responses, and feedback) between a task and an intelligent system over a training phase.

Quantification Using Algorithmic Information Theory

- Algorithmic Information Theory (AIT) may be seen as a computer science extension of Information Theory.

- $H(s)$: The Algorithmic Complexity of a string s, the length of the shortest description of the string in a fixed universal language.
- $H(s_1|s_2)$: Relative Algorithmic Complexity, as the length of the shortest program that, taking s_2 as input, produces s_1.

Quantification Using Algorithmic Information Theory

- Consider
  - A task T
  - Sol_T^{θ}, the shortest of all possible solutions of T of threshold θ (shortest skill program that achieves at least skill θ during evaluation)
  - $TarinSol_{T,C}^{opt}$, the shortest optimal training-time solution given a curriculum (shortest skill program that achieves optimal training-time performance over the situations in the curriculum).

Quantification Using Algorithmic Information Theory

- Generalization Difficulty of a task given a curriculum C and a skill threshold θ

$$GD_{T,C}^{\theta} = \frac{H\left(Sol_T^{\theta} \middle| TarinSol_{T,C}^{opt}\right)}{H\left(Sol_T^{\theta}\right)}$$

- Developer-aware Generalization Difficulty of a task for an intelligent system given a curriculum C and a skill threshold θ

$$GD_{IS,T,C}^{\theta} = \frac{H\left(Sol_T^{\theta} \middle| TarinSol_{T,C}^{opt}, IS_{t=0}\right)}{H\left(Sol_T^{\theta}\right)}$$

Quantification Using Algorithmic Information Theory

- Priors of an intelligent system relative to a task T and a skill threshold θ

$$P_{IS,T}^{\theta} = \frac{H\left(Sol_T^{\theta}\right) - H\left(Sol_T^{\theta} \middle| IS_{t=0}\right)}{H\left(Sol_T^{\theta}\right)}$$

- Experience accrued at step t

$$E_{IS,T,t}^{\theta} = H\left(Sol_T^{\theta} \middle| IS_t\right) - H\left(Sol_T^{\theta} \middle| IS_t, data_t\right)$$

- Experience over a curriculum C

$$E_{IS,T,C}^{\theta} = \frac{1}{H\left(Sol_T^{\theta}\right)} \sum_t E_{IS,T,t}^{\theta}$$

Defining Intelligence

- Intelligence of system IS over scope (sufficient case):

$$I^{\theta_T}_{IS,scope} = \underset{T \in scope}{\text{Avg}} \left[\omega_T \cdot \theta_T \sum_{C \in Cur_T^{\theta_T}} \left[P_C \cdot \frac{GD^{\theta_T}_{IS,T,C}}{P^{\theta_T}_{IS,T} + E^{\theta_T}_{IS,T,C}} \right] \right]$$

- Intelligence of system IS over scope (optimal case):

$$I^{opt}_{IS,scope} = \underset{T \in scope}{\text{Avg}} \left[\omega_{T,\Theta} \cdot \Theta \sum_{C \in Cur_T^{opt}} \left[P_C \cdot \frac{GD^{\Theta}_{IS,T,C}}{P^{\Theta}_{IS,T} + E^{\Theta}_{IS,T,C}} \right] \right]$$

- Schematically, the contribution of each task is: $Expectation \left[\frac{skill \cdot generalization}{priors + experience} \right]$

Key Observations about the Formalism

- A high-intelligence system is one that can generate high-skill solution programs for high generalization difficulty tasks (i.e., tasks that feature high uncertainty about the future) using little experience and priors.
- The measure of intelligence is tied to a choice of scope.
- High skill is not high intelligence: these are different concepts altogether.
- Intelligence must involve learning and adaptation.
- Intelligence is not curve-fitting.
- The measure of intelligence is tied to curriculum optimization.

More than Information Efficiency

- Computation efficiency of skill programs
- Computation efficiency of the intelligent system
- Time efficiency
- Energy efficiency
- Risk efficiency

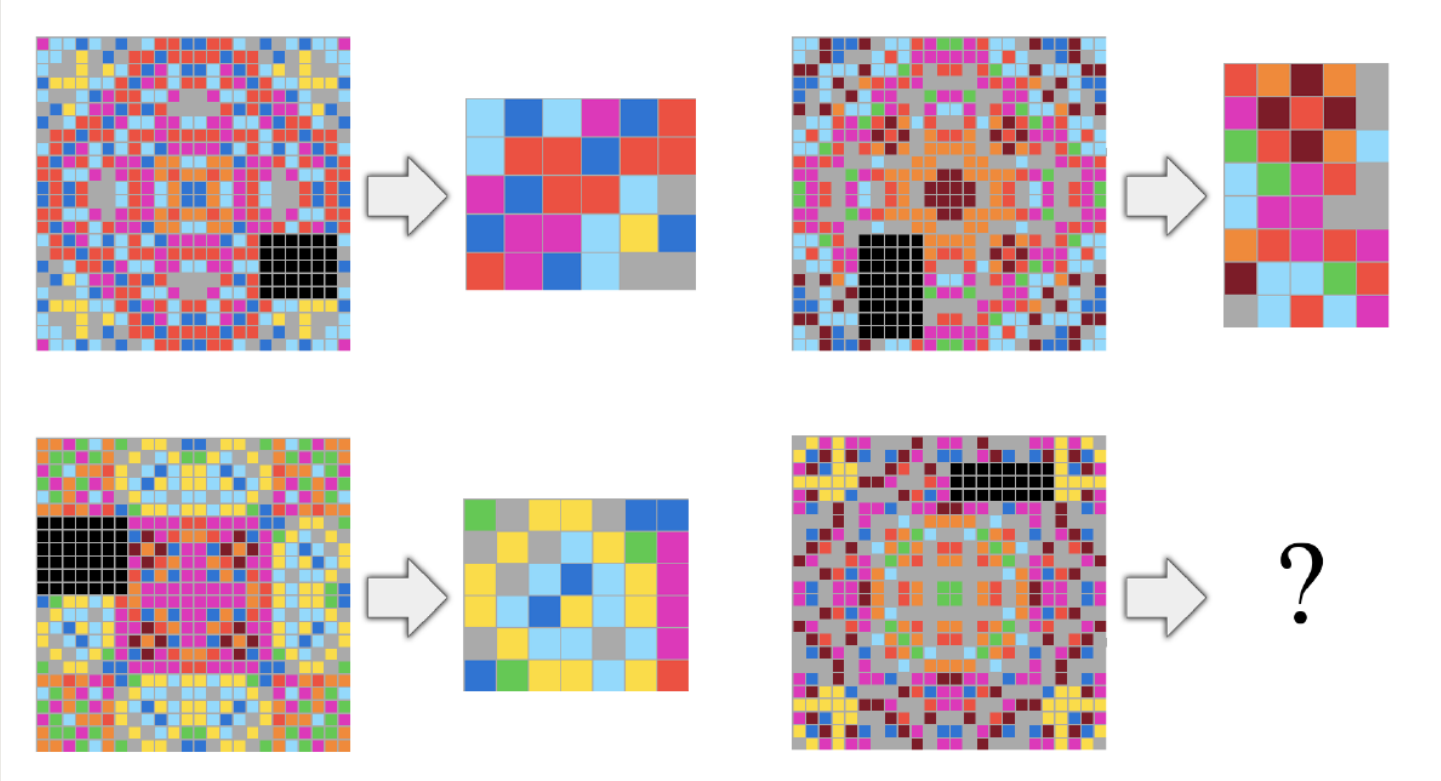
Ideal Intelligence Benchmark

- It should describe its scope of application and its own predictiveness with regard to this scope (i.e., it should establish validity).
- It should be reliable (i.e., reproducible).
- It should set out to measure broad abilities and developer-aware generalization.
- It should control for the amount of experience leveraged by test-taking systems during training.
- It should explicitly and exhaustively describe the set of priors it assumes. The existence of implicit hidden priors may often give an unfair advantage to either humans or machines.
- It should work for both humans and machines, fairly, by only assuming the same priors as possessed by humans (e.g., Core Knowledge) and only requiring a humansized amount of practice time or training data.

A Benchmark Proposal

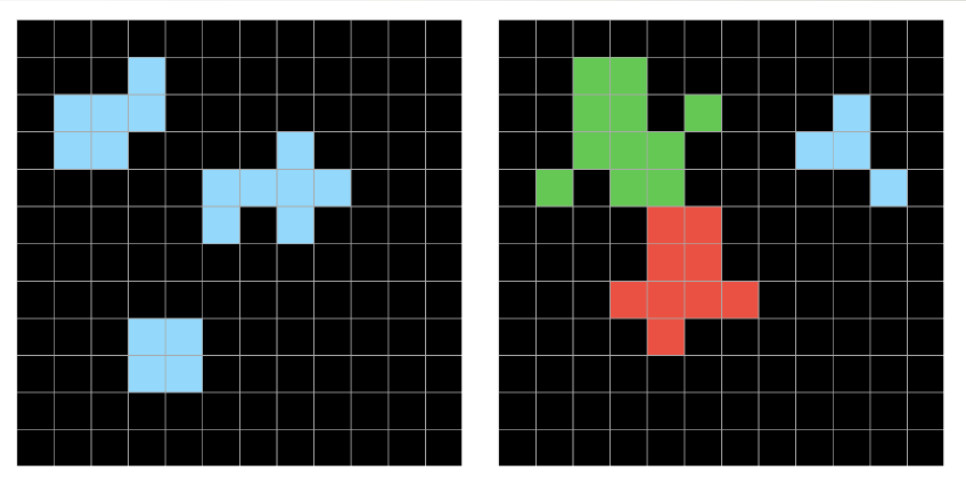
- Abstraction and Reasoning Corpus (ARC)
  - ARC is designed to incorporate as many of the recommendations in this paper as possible.
  - ARC can be seen as a general artificial intelligence benchmark, as a program synthesis benchmark, or as a psychometric intelligence test.
  - ARC comprises a training set and an evaluation set. (https://github.com/fchollet/ARC)
    - The training set features 400 tasks, while the evaluation set features 600 tasks.

Example Task



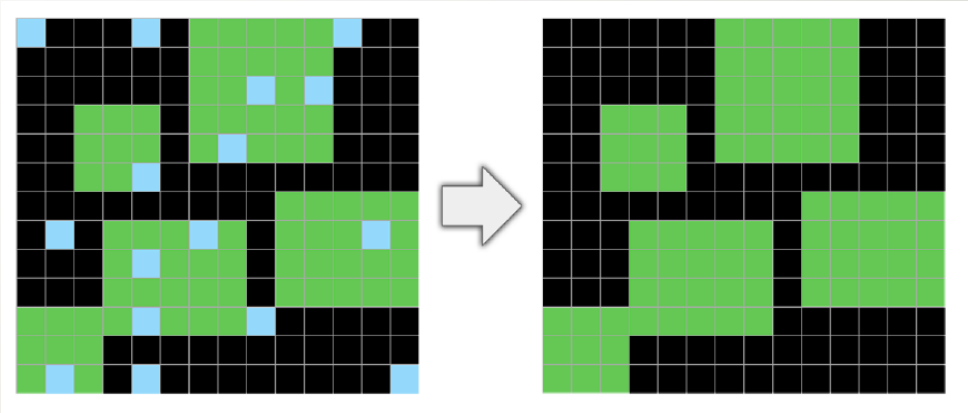
Core Knowledge Priors Assumed by ARC

- Object cohesion
- Ability to parse grids into “objects” based on continuity criteria including color continuity or spatial contiguity, ability to parse grids into zones, partitions.



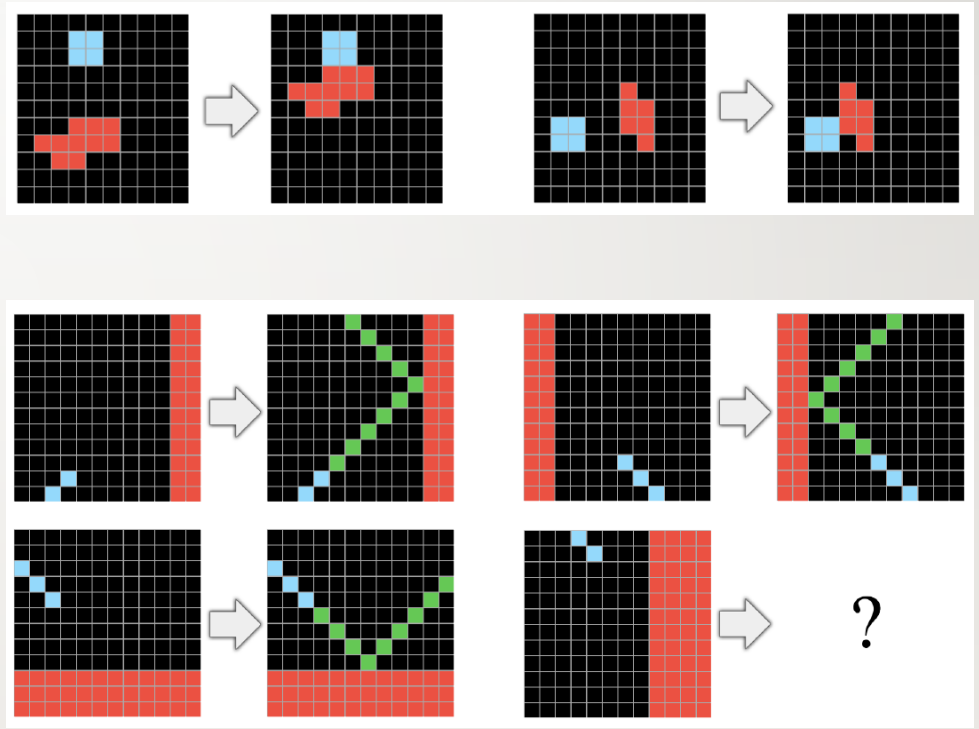
Core Knowledge Priors Assumed by ARC

- Object persistence:
- Objects are assumed to persist despite the presence of noise or occlusion by other objects.



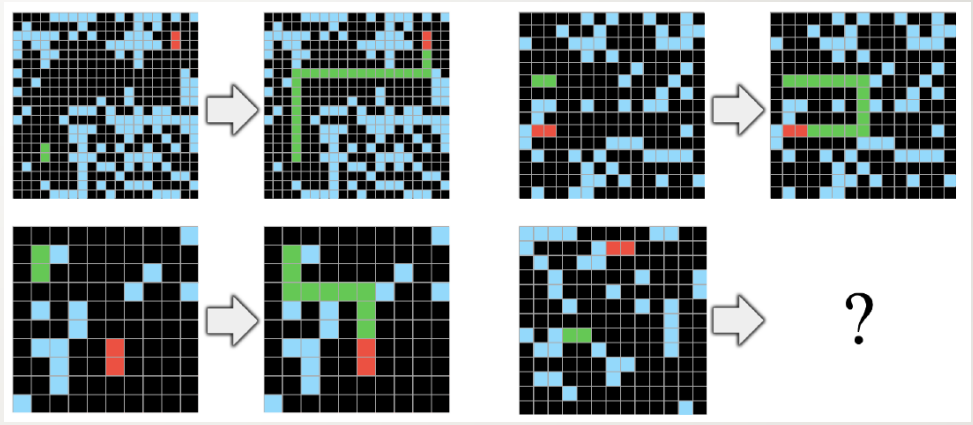
Core Knowledge Priors Assumed by ARC

- Object influence via contact: Many tasks feature physical contact between objects (e.g., one object being translated until it is in contact with another, or a line “growing” until it “rebounds” against another object.



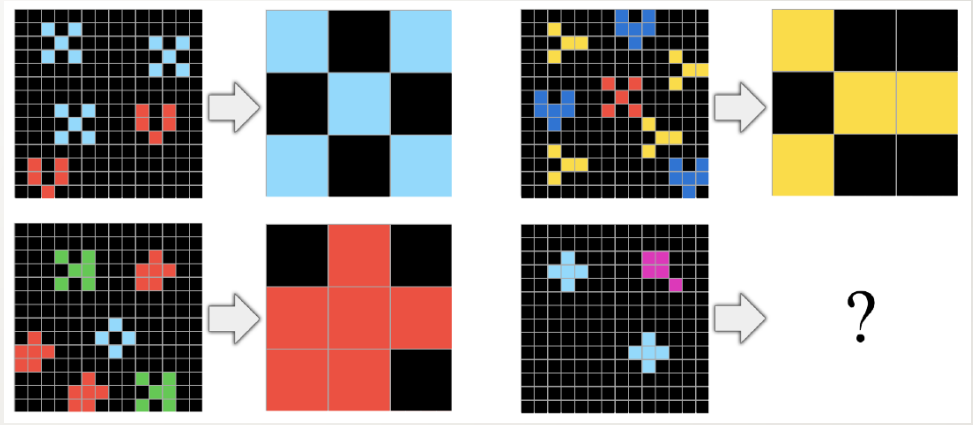
Core Knowledge Priors Assumed by ARC

- Goal-directedness prior
- While ARC does not feature the concept of time, many of the input/output grids can be effectively modeled by humans as being the starting and end states of a process that involves intentionality.



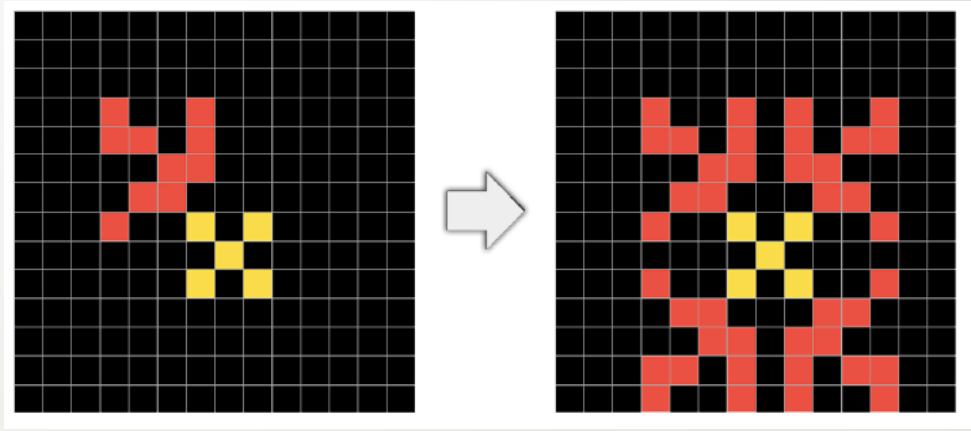
Core Knowledge Priors Assumed by ARC

- Numbers and Counting priors
- Many ARC tasks involve counting or sorting objects (e.g., sorting by size), comparing numbers (e.g., which shape or symbol appears the most? The least? The same number of times? Which is the largest object? The smallest? Which objects are the same size?) or repeating a pattern for a fixed number of time.



Core Knowledge Priors Assumed by ARC

- Basic Geometry and Topology priors
  - Lines, rectangular shapes.
  - Symmetries, rotations, translations.
  - Shape upscaling or downscaling, elastic distortions.
  - Containing / being contained / being inside or outside of a perimeter.
  - Drawing lines, connecting points, orthogonal projections.
  - Copying, repeating objects.



Differences with Psychometric Intelligence Tests

- Unlike some psychometric intelligence tests, ARC is not interested in assessing crystallized intelligence or crystallized cognitive abilities. ARC seeks to only involve knowledge that stays close to Core Knowledge priors.
- The tasks featured in the ARC evaluation set are unique and meant to be unknown to developers of test-taking systems.
- ARC has greater task diversity than typical psychometric intelligence tests.
- ARC tasks are in majority not programmatically generated.

A Hypothetical ARC Solver

- Start by developing a domain-specific language (DSL) capable of expressing all possible solution programs for any ARC task.
- Given a task, use the DSL to generate a set of candidate programs that turn the inputs grids into the corresponding output grids.
- Select top candidates among these programs based on a criterion such as program simplicity or program likelihood.
- Use the top three candidates to generate output grids for the test examples.

Weaknesses and Future Refinements

- Generalization is not quantified.
- Test validity is not established.
- Dataset size and diversity may be limited.
- The evaluation format is overly close-ended and binary.
- Core Knowledge priors may not be well understood and may not be well captured in ARC.

Any Comments?