Playgrounds for Abstraction and Reasoning

Subin Kim [1] Prin Phunyaphibarn [1] Donghyun Ahn [1] Sundong Kim [2 3]

[1]School of Computing, KAIST [2]AI Graduate School, GIST [3]formerly at Data Science Group, IBS

Mini-ARC: Collecting 150 Tasks for Measuring Intelligence

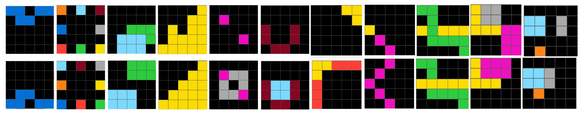


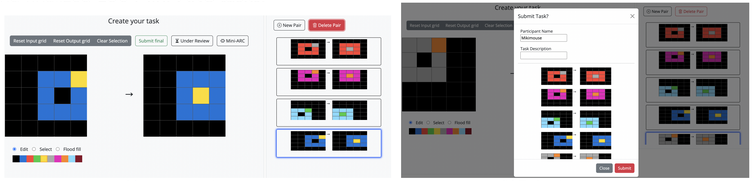
Figure 1. The Mini-ARC dataset is a condensed version of the ARC problem with a 5x5 grid size.

Principles of Curating Mini-ARC Tasks

The primary reason to curate the compact **5x5** data set is to reduce the modeling budget.

We invited 25 colleagues for 4 hours to generate novel 5x5 tasks, including at least four input-output pairs, and instructed to build a task with a clear and unique solution—Figure 2(a). Then, the generated tasks were submitted to the administrator for approval—Figure 2(b).

We pruned around 100 suggested tasks that shared similar concepts during the verification phase and finalized **150 Mini-ARC tasks**.

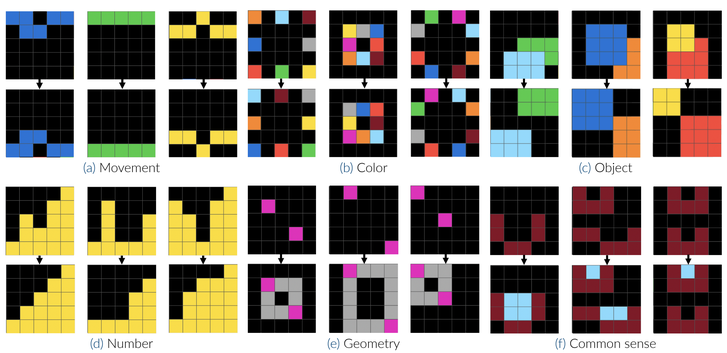


(a) Task generation screen (b) Task submission screen

Figure 2. Interface for curating the Mini-ARC dataset.

Six Categories

- **Movement** tasks are based on dynamic modifications such as flip, rotation, and sliding sideways.
- **Color** tasks are highly dependent of the color aspect of each pixel, such as swapping colors.
- **Object** tasks are dependent to the movement of the object or agent, where an object refers to an area that can be intuitively distinguished from the background.
- **Number** tasks count something, such as the number of pixels of the same color.
- **Geometry** tasks include problems that require the concept of geometric structures.
- **Common-sense** tasks, like maze-pathfinder or Tetris, require high-level induction even though they may be intuitively evident to people.



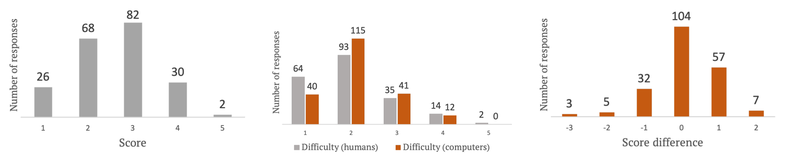
(a) Movement (b) Color (c) Object

(d) Number (e) Geometry (f) Common sense

Figure 3. Representative Mini-ARC examples for each category

Quality Evaluation

In order to rate the novelty and difficulty of the task, participants analyze the created data set and assign a score between 1 and 5. Two grading scales are used, one for human solvers and the other for building AI models. We received 208 answers.



(a) Novelty of tasks (b) Difficulty of tasks (c) Difference between the difficulty for human and computers

Figure 4. Survey results of Mini-ARC

- Contributions from 25 participants allowed us to construct Mini-ARC tasks as sufficiently original problems (Figure 4(a), average 2.59).
- Most of the respondents evaluated that they could solve a given problem intuitively (Figure 4(b), average 2.02).
- Participants felt that developing a program to solve each problem was more difficult than solving it directly (Figure 4(c), average 2.12).
- 64 out of the 208 responses (31%) claimed that implementing the program was more difficult than solving the task manually (Figure 4(c)). 104 respondents (50%) claimed that implementing the program is as difficult as solving the task.

O2ARC: Tools for Collecting Expert Demonstrations

Object-oriented ARC (O2ARC), a browser-based interface, was designed so that participants could solve ARC and Mini-ARC problems.

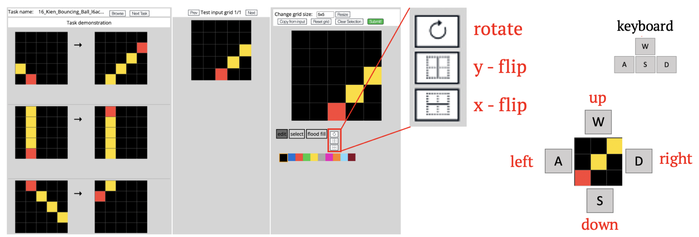
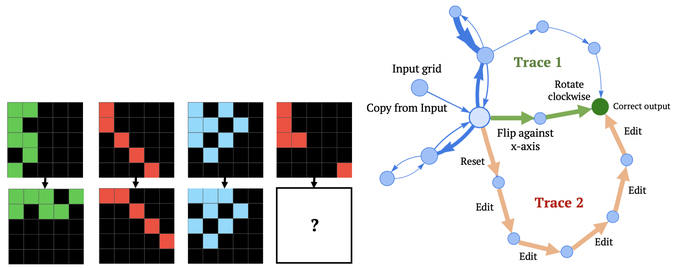


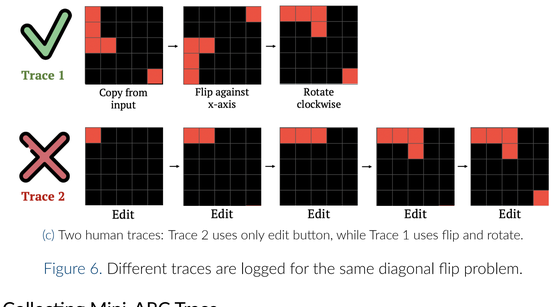
Figure 5. O2ARC tool includes multiple functions to solve Mini-ARC problems.

- O2ARC has added **up, down, left, right, cut, reverse, flip, rotate, and simultaneous selection** to assist the problem-solving process on top of the basic tools provided by ARC creators.
- It is possible to track the combination and order of primitives used in the problem-solving process through O2ARC, which is utilized when collecting Mini-ARC Trace.
- **Interface Design** We tried to show the training input-output pairs one by one instead of showing all the pairs together to collect fine-grained user trajectories that can help us measure the task's difficulty or the wit and agility of the participants. However, this function were commented that they not very useful and was disabled in the final version of O2ARC.

Mini-ARC Trace: Compiled Expert Demonstrations



(a) Task: Diagonal flip (b) State-space graph



(c) Two human traces: Trace 2 uses only edit button, while Trace 1 uses flip and rotate.

Figure 6. Different traces are logged for the same diagonal flip problem.

Principles of Collecting Mini-ARC Trace

- Tasks used for gathering Mini-ARC trace were picked uniformly from six categories.
- 20 participants were gathered to solve ten of Mini-ARC tasks within two hours, where each set contains five tasks, one from each category.
- Collecting traces of common-sense problems has been postponed. Those tasks requires a fairly original solution, thus traces collected from these problems are unlikely to be applied to solve other types of problems.
- If the participant submits three consecutive wrong answers, the system determines that the user cannot find the solution to the problem.

For each Mini-ARC task, we combined all traces in a state-space graph—Figure 6(b). Each node represents the state of the output grid, and the green node represents the correct output grid.

Challenges and suggestions

Challenges As seen in Figure 6(b)–(c), not all traces contain the high-level process of human intuition. Trace 1 reflects the intuition of diagonal flips using compact movements, and this solution can be generalized to the other pairs following the same rules, while Trace 2 cannot. The restricted input and output grid size of the Mini-ARC tasks may take part in increasing naive actions. How to gather traces that reflect intuition is still an open question.

Suggestions The purpose of the Mini-ARC trace is to collect reference trajectories of the Mini-Arc tasks. The following are suggestions for utilizing Mini-ARC in future research.

- Utilize the traces as a replay buffer for training agents through imitation learning.
- Analyze the traces based on action sequences to find new primitives that can be generally used for solving multiple Mini-ARC tasks.