Thesis for Bachelor's Degree

O2ARC 3.0: A Platform for Solving and Creating ARC Tasks

Suyeon Shim

School of Electrical Engineering and Computer Science

Gwangju Institute of Science and Technology

2025

학사학위논문

O2ARC 3.0: ARC 문제 풀이 및 생성 플랫폼

심수연

전기전자컴퓨터공학부

광주과학기술원

2025

O2ARC 3.0: A Platform for Solving and Creating ARC Tasks

Advisor: Sundong Kim

by

Suyeon Shim

School of Electrical Engineering and Computer Science

Gwangju Institute of Science and Technology

A thesis submitted to the faculty of the Gwangju Institute of Science and Technology in partial fulfillment of the requirements for the degree of Bachelor of Science in the Electrical Engineering and Computer Science Concentration

Gwangju, Republic of Korea

Dec 06, 2024

Approved by

Professor Sundong Kim

Committee Chair

O2ARC 3.0: A Platform for Solving and Creating ARC Tasks

Suyeon Shim

Accepted in partial fulfillment of the requirements for

the degree of Bachelor of Science

Dec 06, 2024

Committee Chair _____

Prof. Sundong Kim

Committee Member _____

Prof. Mansu Kim

Dedicated to my family.

BS/EC 20215117 Suyeon Shim(심수연). O2ARC 3.0: A Platform for Solving and Creating ARC Tasks (O2ARC 3.0: ARC 문제 풀이 및 생성 플랫폼). School of Electrical Engineering and Computer Science. 2025. 27p. Advisor Prof. Sundong Kim.

Abstract

I introduce the O2ARC 3.0 interface for the Abstraction and Reasoning Corpus (ARC). O2ARC 3.0 gamifies the experience, fostering user engagement through competitive features and community-driven problem creation and evaluation. Built with a React frontend and NestJS backend, the platform provides a responsive and intuitive interface for efficient rule inference. This approach not only improves data collection for AI training but also enhances the problem-solving process, offering a scalable solution for advancing cognitive AI research. O2ARC is available at `https://o2arc.com`.

BS/EC 20215117 심수연. O2ARC 3.0: ARC 문제 풀이 및 생성 플랫폼. 전기전자컴퓨터공학부. 2025. ??p. 지도교수: 김선동 교수님.

국 문 요 약

본 연구에서는 Abstraction and Reasoning Corpus (ARC)를 위한 O2ARC 3.0 인터페이스를 소개합니다. O2ARC 3.0은 게이미피케이션 요소를 도입하여 경쟁적인 기능과 커뮤니티 주도 문제 생성 및 평가를 통해 사용자 참여를 유도합니다. React 기반 프론트엔드와 NestJS 백엔드로 구축된 이 플랫폼은 효율적인 규칙 추론을 위한 반응형이고 직관적인 인터페이스를 제공합니다. 이러한 접근 방식은 AI 훈련을 위한 데이터 수집을 개선할 뿐만 아니라 문제 해결 과정도 강화하여 인지 AI 연구를 발전시키기 위한 확장 가능한 솔루션을 제시합니다. O2ARC는 `https://o2arc.com`에서 확인할 수 있습니다.

Contents

List of Figures

Chapter 1

Introduction

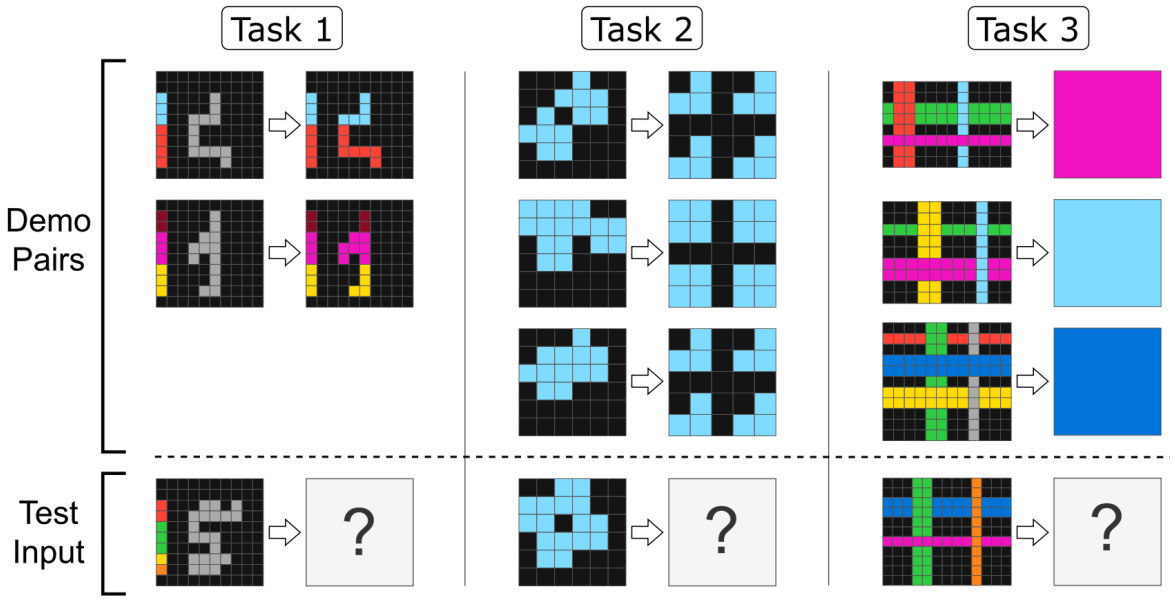


Figure 1.1: Three different ARC tasks. The ARC dataset is designed to assess the generalization and learning abilities of AI systems across novel tasks, which is crucial for measuring true intelligence.

The Abstraction and Reasoning Corpus (ARC), introduced by François Chollet (1)chollet2019ARC, presents grid-based tasks that target both humans and AI systems to infer rules from given input-output pairs and apply them to a test input. This task demands abductive reasoning skills and a level of common sense—capabilities where human cognition excels. In contrast, current AI systems struggle, achieving no more than 30% accuracy on these tasks (2; 3). To bridge this gap, researchers have highlighted the need for interfaces to collect additional tasks (4) and to

understand the human-solving process (5; 6). Building upon the original interface (7), tools like Language-Annotated ARC (8), Lab42's ARCreate (9), and Object-Oriented ARC (O2ARC 1.0) (6) have been developed for this purpose.

In response, I present O2ARC 3.0, an evolution of the tool that gamifies the problem-solving experience. O2ARC 3.0's interface is designed to mimic the engagement of puzzle games, encouraging voluntary involvement and diminishing the need for external rewards. I have added competitive elements like leaderboards and a system for creating and peer-reviewing tasks to enhance user engagement. The interface is designed to discourage inefficient single-pixel editing and to promote object-based manipulations, ensuring the generation of high-quality human traces.

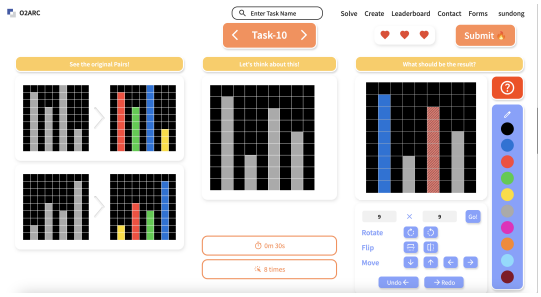
Enhancements of O2ARC 3.0 over Previous ARC Tools O2ARC advances the capabilities of existing ARC problem-solving tools by offering a broader set of operations for user interaction. It incorporates a react frontend with a nestJS backend architecture, optimized for performance and scalability. The implementation of a scoring mechanism linked to leaderboards has been instrumental in increasing user engagement, as indicated by user studies that reported motivation without financial incentives.

O2ARC addresses the limitations observed in prior tools where users frequently defaulted to single-pixel painting. The redesigned user interface encourages the use of a variety of operations, thereby facilitating an approach that better reflects the complex reasoning ARC problems aim to measure. The platform provides immediate feedback on problem-solving efficiency and introduces an IQ score that is revealed after solving a

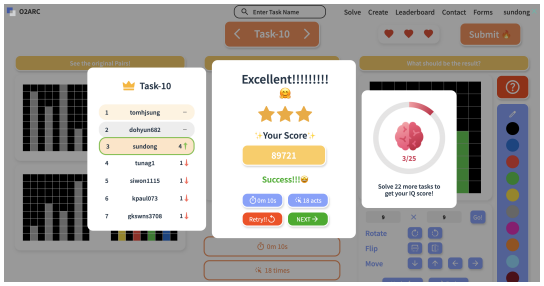
predetermined number of problems. This score increases with the number of problems solved, promoting long-term engagement and data collection for AI benchmarking.

Chapter 2

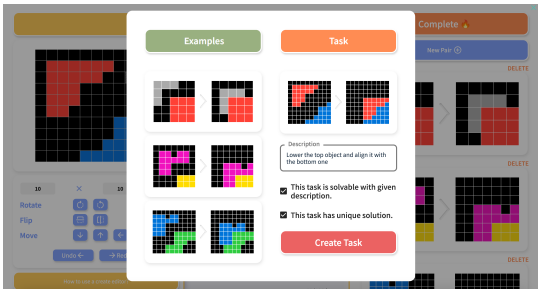
Functionalities and User Engagement



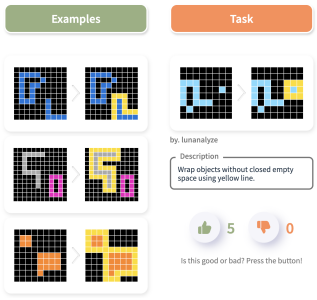
(a) Solve a task with editor panel.



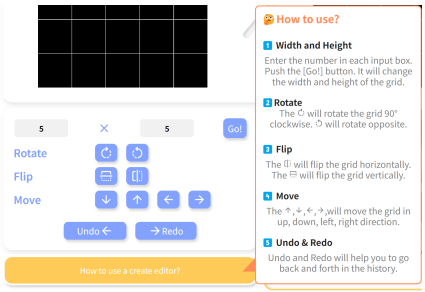
(b) Check your score after solving the task.



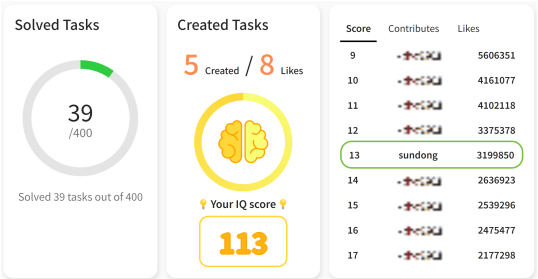
(c) Create a task with four or more pairs.



(d) Evaluate the tasks that others have created.



(e) View the instructions by pressing the '?' button.



(f) See your achievements in the profile tab.

Figure 2.1: Content that users experience in the O2ARC tool.

Within O2ARC 3.0, users can engage in various activities:

Solving ARC Tasks The platform allows users to engage with a random task via the navigation bar and presents subsequent tasks in sequence. The editor panel offers various operations, such as rotate, flip, resize grid, move, undo, and redo, to facilitate the construction of the predicted output. Incorrect attempts reduce the user's 'lives', whereas correct solutions display task-specific scores and rankings on the leaderboard, rewarding efficiency and accuracy. To foster continued participation, an IQ revelation feature is unlocked after 25 tasks are completed. Trial and time restrictions are in place to deter excessive focus on any single task.

Creating ARC Tasks Users can construct their tasks, requiring a minimum of three demonstrations and a test input-output pair for submission. The creation process mirrors the solving interface, with additional functionalities like input-output reset. Before submission, creators verify the uniqueness of their solutions and the adequacy of the accompanying descriptions. Community members can then assess the quality of these tasks on the 'created' page.

Profile and Leaderboard The profile tab provides users with a personalized summary of their problem-solving activity, likes received, and leaderboard position. The leaderboard dynamically updates to reflect user scores and rankings, encouraging competition with visual animations and offering recognition for both problem-solving proficiency and creative contributions.

Chapter 3

System Design

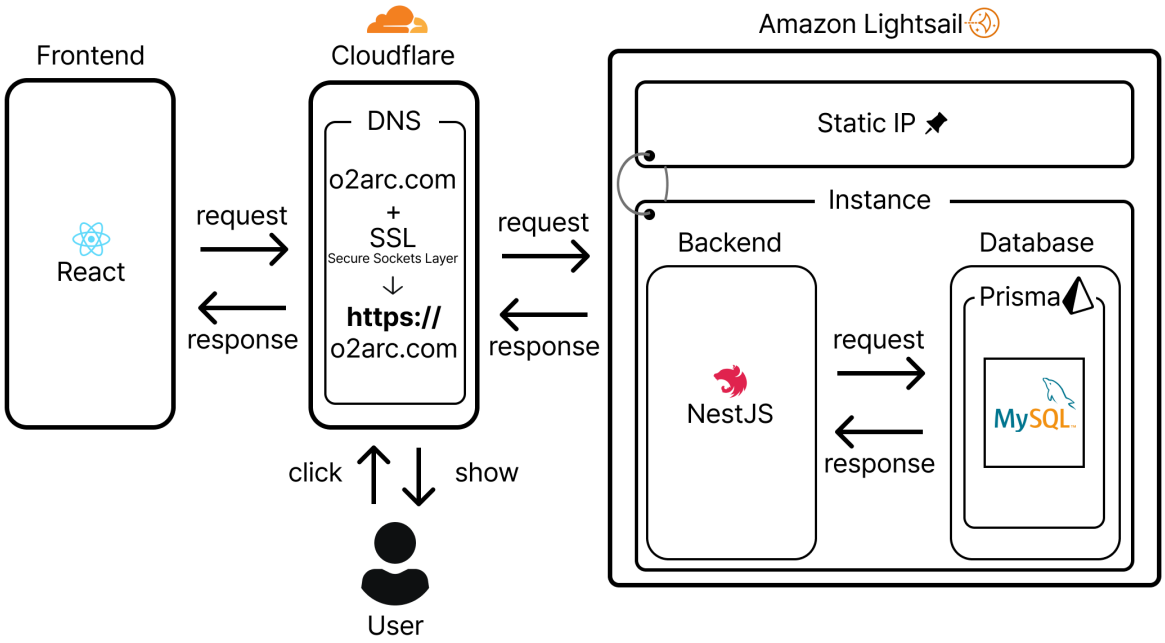


Figure 3.1: System design of the O2ARC 3.0.

O2ARC 3.0 uses React and NestJS to build the frontend and backend, with the frontend hosted on Cloudflare and the backend hosted on AWS LightSail. Figure **??** illustrates the O2ARC 3.0 system design.

Frontend (React) The frontend of O2ARC 3.0 utilizes React, a leading frontend framework, to create seamless user interfaces. By employing a Single Page Application (SPA) architecture alongside frontend caching, the application delivers an immersive experience with minimal loading times. The use of TypeScript, combined with a func-

tional React architecture, aids in simplifying maintenance and ensuring robustness. Operations within the editor grid are handled locally to reduce backend communication delays. Additionally, the application features dynamic animations on both the leaderboard and scoreboards to maintain user interest and engagement. I merged and modulized JS files into a single JS file with Vite.

Backend (NestJS) NestJS (10) is a Node.js framework used for building server-side applications. Also, it supports Express, providing an HTTP server. O2ARC 3.0 utilized it for the backend, with code written in TypeScript.

NestJS provides a uniform structure consisting of a controller, a service, a repository, and a module, which makes the code easier to understand and maintain. Additionally, the service file defines which repository functions to call. A service function can call multiple repository functions. The repository file utilizes the Prisma service to perform operations on a specific table in the database, such as finding, inserting, and deleting rows.

Additionally, Swagger is a tool that facilitates communication between the frontend and backend. O2ARC 3.0 includes a Swagger component that generates API documentation for the frontend in the form of a Swagger UI.

Calculation logic was designed to analyze data collected from users and update the database accordingly; a representative example of this logic is as follows.

The diagram in Figure 3.2 illustrates the process flow for managing user responses, scoring, and leaderboard updates in a problem-solving platform. The process is initiated when a **User Log** is received, encompassing actions such as checking the answer,

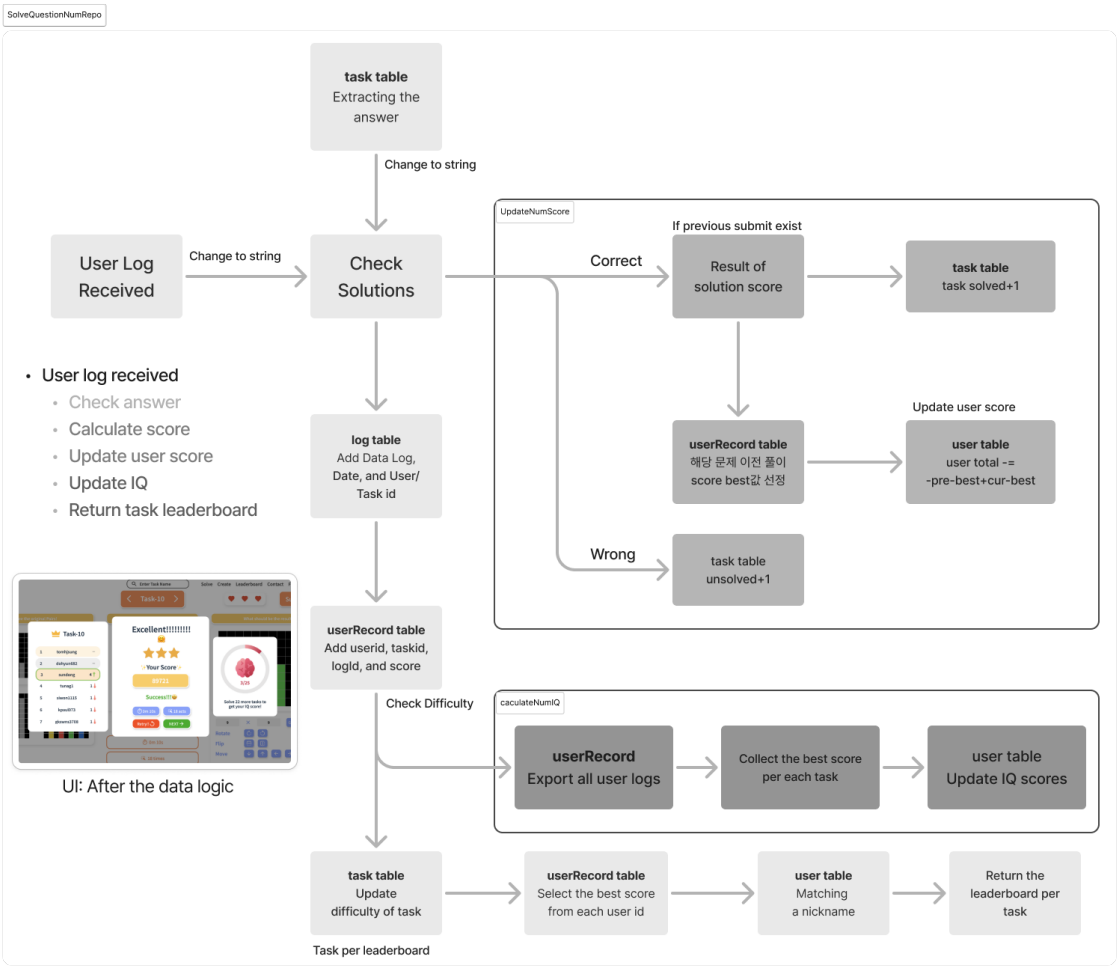


Figure 3.2: Process Flow for Solution Checking, Scoring, and Leaderboard Management in the User Engagement System

calculating the score, updating the user's overall score and IQ, and preparing the leaderboard for the specific task.

- **Check Solutions**: Upon receiving the user log, the system converts the answer into a string format and checks it against the correct answer stored in the *task table*. This table is used to extract the correct answer and track the status of each task (solved or unsolved).

- **Log and Score Update**:

  - If the solution is correct, the *log table* records details such as the log date and user/task ID, and the *userRecord table* updates the user's best score for that particular task if it's higher than any previous attempt. The *task table* is updated to increment the solved count.

  - If the solution is incorrect, the system logs the attempt, updates the *task table* to increment the unsolved count, and leaves the user score unchanged.

- **User Score Calculation**: For correct answers, the *user table* is updated to reflect the user's cumulative score. This is calculated by subtracting the previous best score for the task and adding the new best score, ensuring that only the highest score for each task contributes to the user's total.

- **Task Difficulty Adjustment**: The system uses the user log data to assess task difficulty, which is periodically updated in the *task table* to maintain accurate difficulty ratings based on user performance.

- **Leaderboard and IQ Score Calculation**: The *userRecord table* exports all user logs and determines the best scores per task for each user, which are then compiled to update IQ scores in the *user table*. After updating, the leaderboard is generated, matching each user ID to their nickname for display purposes.

- **User Interface**: Finally, the updated scores, task statuses, and leaderboard rankings are reflected on the user interface, providing users with feedback on their performance and their standing relative to others.

Database (Prisma and MySQL) Prisma is an ORM (Object Relational Mapping) framework. It automatically converts the Prisma schema syntax, which contains the database structure, into MySQL statements and executes them. The O2ARC 3.0 database was accessed through TablePlus, which supports creating or dropping tables and executing queries.

The database stores refined logs collected from users, organizing the data into various tables. The following outlines the relationships and components of the primary tables.

Figure 3.3 presents the database schema designed to manage user interactions, task evaluation, and content organization on the platform. This schema comprises six primary tables: *User*, *Task*, *Content*, *Task Evaluation*, *Log*, and *User Record*. Each table serves a distinct role in capturing user actions, storing task data, and updating performance records.

- **User Table**: This table stores basic user information, including a unique identi-

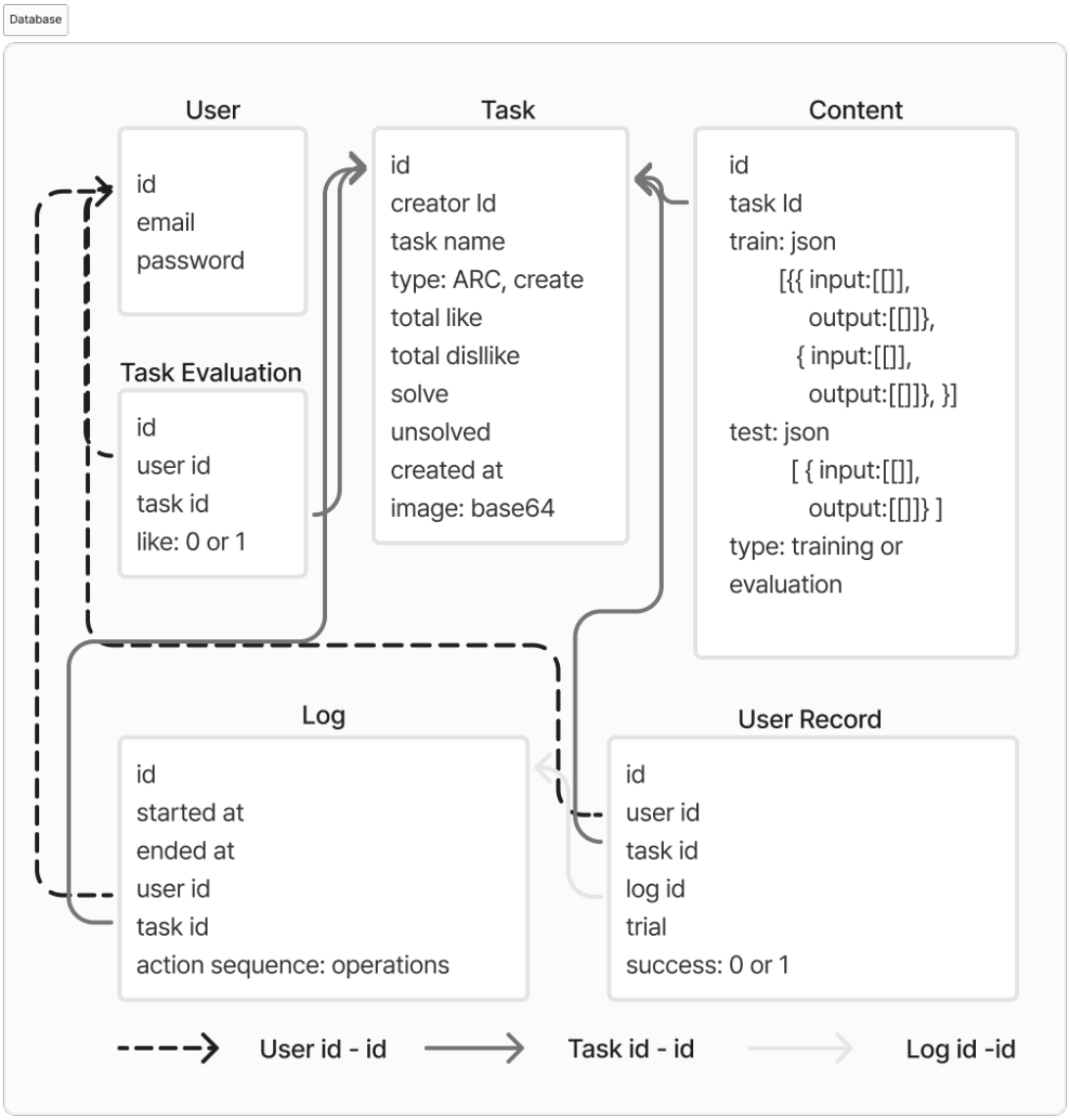


Figure 3.3: Database Schema for User Interaction, Task Evaluation, and Content Management

fier (*id*), email, and password. It serves as the core entity linking users to their interactions with tasks and evaluations.

- **Task Table**: The *Task* table defines each task available on the platform, containing fields such as *id*, *creatorId*, *taskName*, *type* (e.g., ARC or create), and metadata like *totalLike*, *totalDislike*, *solve*, *unsolved*, *createdAt*, and an *image* in Base64 format. This table tracks task characteristics and user engagement metrics, such as total likes and dislikes, as well as the number of solved and unsolved attempts.

- **Content Table**: The *Content* table holds the task content in JSON format, including both *train* and *test* data for each task, consisting of input-output pairs for training or evaluation purposes. The *type* field distinguishes between training and evaluation content, allowing the platform to serve appropriate tasks to users based on their interaction phase.

- **Task Evaluation Table**: This table links user feedback to specific tasks, storing *id*, *userId*, *taskId*, and a *like* field (0 or 1) indicating whether the user liked or disliked the task. This data supports analytics on task popularity and user preferences.

- **Log Table**: The *Log* table records each user's activity session, including fields such as *id*, *startedAt*, *endedAt*, *userId*, *taskId*, and an *actionSequence* that captures the sequence of operations performed during the session. This table provides a comprehensive log of user behavior, allowing for detailed analysis of interaction

patterns.

- **User Record Table**: The *User Record* table tracks the outcome of each user's attempts at solving tasks. It contains fields such as *id*, *userId*, *taskId*, *logId*, *trial*, and *success* (0 or 1), indicating whether the attempt was successful. This table is essential for tracking user performance, enabling the platform to monitor progress and calculate scores or IQ levels for each user.

The schema reflects the relationships between users, tasks, and logs:

- **User-Task Relationship**: The *Task Evaluation* table links user preferences to specific tasks, while the *Log* and *User Record* tables capture detailed user interactions and outcomes with tasks.

- **Task Content Management**: The *Content* table provides structured data for each task, supporting both training and evaluation phases.

- **User Performance Tracking**: User success and trial counts are maintained in the *User Record* table, enabling the calculation of user scores and progression metrics.

This schema enables efficient management of user interactions, task evaluation, and content organization, facilitating robust analytics and continuous improvement of the platform's engagement and task difficulty calibration.

Backend Hosting and Storage (Amazon Lightsail) Amazon Lightsail is a cloud computing service that provides a virtual server (instance) for backend hosting. O2ARC

3.0 tools utilized a Lightsail instance with a Linux/Unix platform and Ubuntu operating system. The instance has 2GB of RAM, 2 vCPUs, and a 60GB SSD. Pins a static IP to the instance so that it always uses the same IP even if the instance is restarted. It contains the backend system and database tables, where ARC task data and user-solving logs are stored.

Https Hosting (Cloudflare) Cloudflare is a CDN service, specialized in deploying static frontends. When a user accesses o2arc.com through a browser, Cloudflare delivers the prepared frontend build files to the browser, which then interprets them to render O2ARC's UI and communicate with the Backend. Additionally, I have integrated Cloudflare with our GitHub repository to enable automatic deployment for each branch, automating the tasks necessary for deployment.

Design System (Figma) The UI for each page of the O2ARC 3.0 tool was designed using Figma, a collaborative interface design tool that provides various design actions. Figma also offers a developer mode that is useful for building UI on the frontend.

Chapter 4

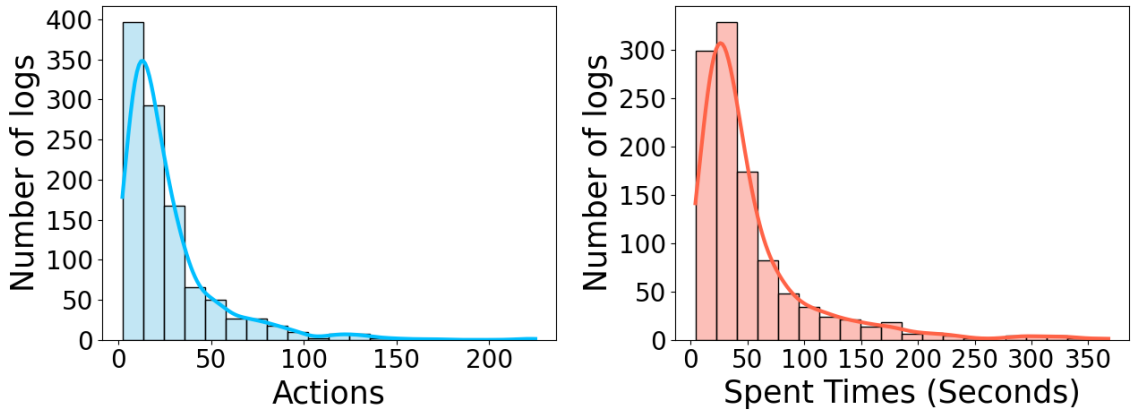
User Engagement and Impact

UX Design In our UX design, I prioritize object-oriented operations over single-pixel manipulations to better reflect human problem-solving strategies. The interface initially featured Edit, Select, and Flood-Fill modes (7), each toggled via specific buttons. To minimize single-pixel edits, I removed the Edit mode and modified the Flood-Fill mode to activate through a double-click, now named Flood Selection. This adjustment inherently encourages users to adopt object-oriented operations by limiting the granularity allowed by the previous Edit mode and making object manipulation more accessible.

A key feature of the improved tool is the interaction between participants to stimulate engagement. In the previous version, participants solved one static problem each without interacting with each other, but in this version, they were given a score based on the time spent solving the problem and the number of actions, which was used to create a leaderboard for each problem and an overall score leaderboard. By bringing people together offline to solve problems, I found that users solved an average of n problems in an hour, which is encouraging when compared to Johnson's (Flex and flexible) results where users were monetarily rewarded for solving problems, and suggests that there is potential to collect large-scale solving data at low cost. I also implemented a problem creation system that allows participants to check each other's work and give recommendations if they think it's good, or disapproval if they think it's bad. I added

a leaderboard that shows the number of likes received, similar to a problem score. As a result, we've seen an improvement in the quality of questions users are creating, and we've reduced the amount of work it takes to review questions (and quantify them, if possible). This also suggests that it is possible to augment ARC data at scale at a low cost.

User Study A user study involving 50 participants evaluated the solve and create functions of O2ARC 3.0, with 24 providing detailed feedback. The tool received high satisfaction ratings (Likert scale from 1–10): 8.7 overall, 8.6 for solving, 8.7 for creating, and 8.8 for evaluating tasks. Key highlights include the leaderboard's role in enhancing competition and motivation, the tool's ease of use and intuitive design, and its effectiveness in engaging users with ARC concepts. This feedback emphasizes the improvements in user engagement and data collection compared to previous versions.



(a) Distribution of Actions (b) Distribution of Spent Times

Figure 4.1: User study analysis

[ht]

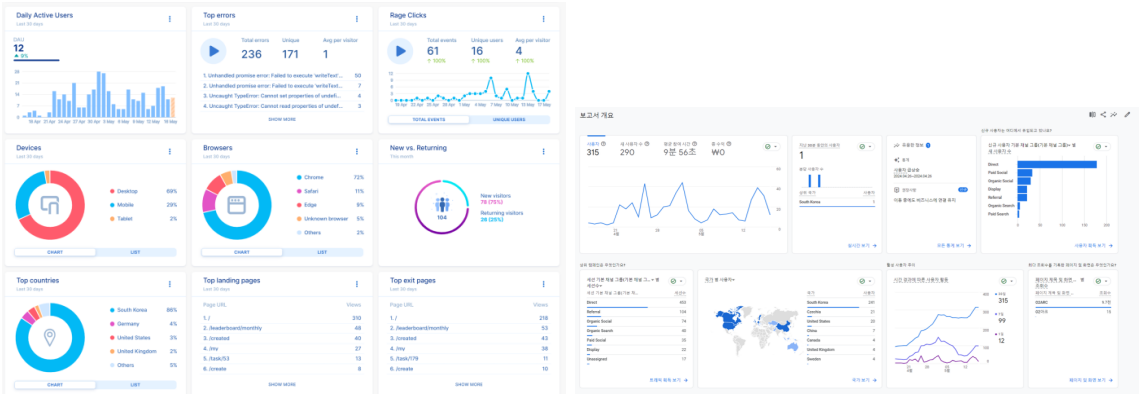
An alpha test was conducted with 50 users. The solve function was tested for

approximately 2 hours and the create function for approximately 1 hour. 24 users provided their evaluation of O2ARC 3.0. The users gave positive feedback, with an average overall satisfaction rating of 8.7 for O2ARC 3.0, 8.6 for the solve page, 8.7 for the create page, and 8.8 for the evaluate page. Users provided specific ratings for features. Here are quotes from their feedback.

"The leaderboard was good for fostering a sense of competition. and motivated me to try harder.", "The leaderboard helped me keep solving problems without thinking." The leaderboard incentivized them to persist in solving problems. Including the Game-Like Component in O2ARC 3.0, as opposed to former versions, facilitated the continuous collection of user data.

"It was good to have easy access to the web", "The web quality was high and intuitive, so it was easy to use.", "The server response is fast and the design is pretty." The high accessibility of the web page lowered the barrier to entry for users, and the usability was well-received. Also, the UI design is attractive to users, making them want to keep using the tool.

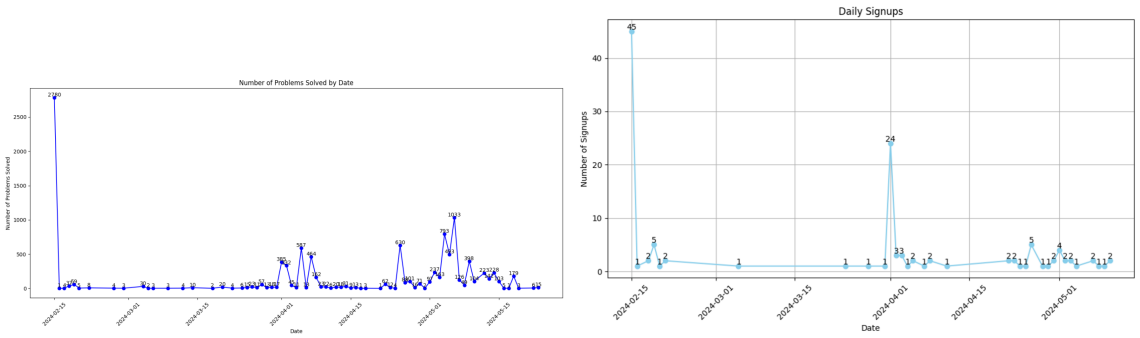
"Easy access to ARCs helped increase interest.", "It was good to learn about ARC AI while discussing with the person next to me.", "It was good to think about how AI solves problems while solving problems." Users appreciated that the tool allowed them to think and learn about ARC and AI while solving tasks. O2ARC 3.0 helped users who didn't know much about ARC to start thinking about ARC. They appreciated the process of accessing and solving difficult tasks together in O2ARC 3.0.



(a) Dashboard of Smartlook (b) Dashboard of Google Analytics

Figure 4.2: User Behavior Analysis

User Analysis This section examines user behavior through the Smartlook and Google Analytics dashboards.

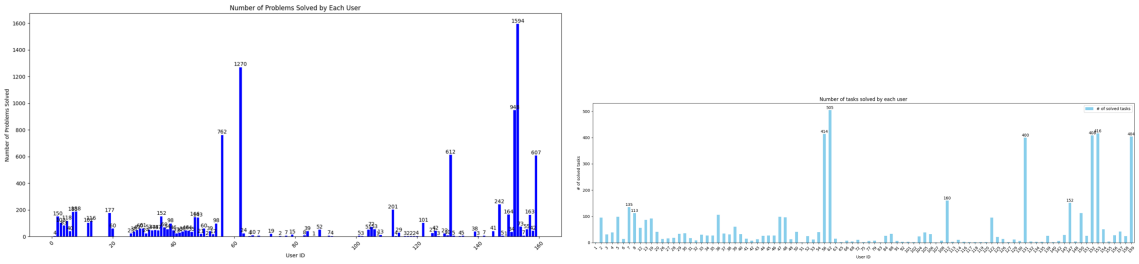


(a) Daily Data Collection Count (b) Daily Sign-up Counts

Figure 4.3: User Activity Metrics: Daily Data Collection and Sign-up Trends

User Engagement and Growth Metrics Over the course of the study, a total of 10,678 data points were collected, with daily statistics displayed in Figure 4.3a. These metrics provide insight into user engagement trends, allowing us to identify active periods and gauge the platform's popularity on a day-to-day basis. Additionally, user sign-up data, shown in Figure 4.3b, highlights critical moments in user acquisition, such as notable spikes around the February 15 lab event and the official launch on April 1.

These events significantly contributed to a total of 161 user sign-ups, demonstrating how strategic events and launches can drive user growth and engagement. Together, these metrics underscore the importance of targeted outreach and the potential for sustained engagement through well-timed promotional activities.



(a) Number of Solutions Submitted by Each User (b) Number of Problems Solved by Each User

Figure 4.4: User Engagement Metrics: Solution Submissions and Problems Solved Per User

User Engagement Through Solution Submissions and Problem Solving Understanding the engagement levels of users, particularly heavy users who actively solve multiple problems, is key to optimizing platform performance. Figures 4.4a and 4.4b illustrate the number of solutions submitted and problems solved per user, providing insights into user behavior patterns. Tracking these metrics not only reveals which users are most engaged but also highlights opportunities to enhance retention by encouraging problem-solving persistence. By analyzing the data, I can tailor the platform to support users at different engagement levels, thereby fostering a more robust and active user community.

Understanding User Interaction Patterns Through Operation Counts To gain insights into how users engage with problem-solving on our platform, I analyzed

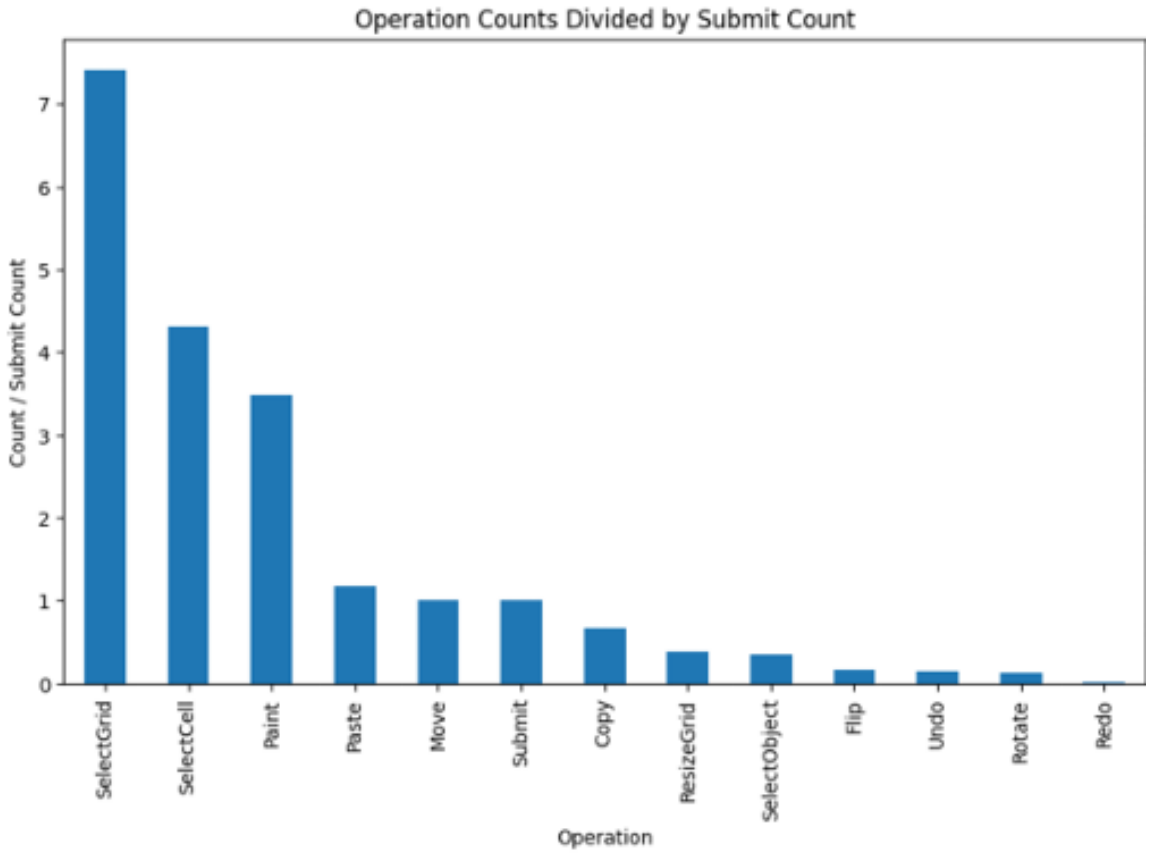


Figure 4.5: Interaction Frequencies: Counts of Operations Normalized by Submit Count

various operations performed during their interactions. As shown in Figure 4.5, the most frequently used operations were *SelectGrid* and *SelectCell*, with *SelectGrid* dominating the interaction. This suggests that users prefer selecting broader sections of the interface rather than targeting individual cells, indicating a tendency to approach problem-solving from an "object-oriented" perspective.

- **Preference for Object-Oriented Interactions**: Users primarily engaged with grid-level selections, followed by cell-level selections. This pattern implies that users view problems as entities or objects rather than discrete cells, aligning with an object-oriented approach to problem-solving. This insight could inform the design of future interaction models, making the interface more intuitive for users inclined to interact at a broader, object-level scale.

- **High Selection and Manipulation Frequency**: The *Selection* operation was the most common, followed by *Object Manipulation* functions such as SelectObject, Flip, and Rotate. This behavior highlights the importance of providing versatile selection tools and flexible object manipulation capabilities to enhance user engagement and problem-solving efficiency.

- **Implications for Reinforcement Learning (RL) Model Design**: The data collected on user interaction patterns can inform the design of reward structures in RL models. Specifically, high engagement with object-based selections suggests that rewards could be structured around efficient object manipulation and selection strategies, encouraging users to develop optimal problem-solving techniques.

The analysis of operation counts provides valuable insights into user behavior, particularly the inclination towards object-oriented problem-solving. By focusing on operations like *SelectGrid* and *SelectObject*, I can enhance the platform's usability to align better with user preferences. Additionally, the observed patterns may inform the design of future problem-solving tasks and interaction models, particularly in optimizing the user experience for educational or training purposes.

Broader Impact Inspired by O2ARC, sister projects like ARCLE (11) are emerging, leveraging its state and action spaces for training reinforcement learning agents. This collaboration bridges human cognitive processes with AI learning, enriching the ecosystem. Through ARCLE, O2ARC's rich dataset becomes a fertile ground for developing AI agents with human-like reasoning skills.

Chapter 5

Conclusion

This thesis presented O2ARC 3.0, a cutting-edge platform that redefines user engagement and data collection for the Abstraction and Reasoning Corpus (ARC). By introducing gamification elements such as leaderboards and peer-reviewed task creation, O2ARC 3.0 transforms the problem-solving experience into an interactive and motivating journey. Its innovative architecture, powered by React and NestJS, ensures a seamless and scalable interface that effectively bridges the gap between human cognition and AI reasoning. This paper introduces O2ARC 3.0, an engaging interface for Abstraction and Reasoning Corpus (ARC). O2ARC 3.0 is accessible at `https://o2arc.com`.

The platform has demonstrated measurable success through high user satisfaction ratings and increased engagement metrics, showcasing its potential to revolutionize ARC-related research. O2ARC 3.0 not only improves the quality of user-generated tasks but also reduces the cost and effort required for large-scale data collection, providing a valuable resource for advancing cognitive AI systems.

Beyond its immediate applications, O2ARC 3.0 sets the stage for future advancements in reinforcement learning and human-AI collaboration, as seen in its influence on sister projects like ARCLE. These contributions underline the platform's role as a foundational tool for the development of AI agents capable of human-like reasoning.

Looking forward, further refinement of task diversity, user interaction models, and

adaptive AI integration could unlock even greater possibilities. By continually evolving, O2ARC 3.0 holds the promise of not only advancing AI research but also inspiring a deeper understanding of human problem-solving strategies. It is a step toward a future where human and machine intelligence collaboratively tackle the complexities of the world.

References

[1] F. Chollet, "On the Measure of Intelligence," *arXiv:1911.01547*, 2019.

[2] Lab42, "Arcathon leaderboard." `https://lab42.global/arcathon/leaderboard/`, 2024.

[3] M. Bober-Irizar and S. Banerjee, "Neural Networks for Abstraction and Reasoning: Towards Broad Generalization in Machines," *arXiv:2402.03507*, 2024.

[4] A. Moskvichev, V. V. Odouard, and M. Mitchell, "The ConceptARC benchmark: Evaluating Understanding and Generalization in the ARC Domain," *TMLR*, 2023.

[5] A. Johnson, W. K. Vong, B. M. Lake, and T. M. Gureckis, "Fast and Flexible: Human Program Induction in Abstract Reasoning Tasks," in *CogSci*, 2021.

[6] S. Kim, P. Phunyaphibarn, D. Ahn, and S. Kim, "Playgrounds for Abstraction and Reasoning," in *NeurIPS Workshop on Neuro Causal and Symbolic AI*, 2022.

[7] F. Chollet, "ARC Testing Interface." `https://github.com/fchollet/ARC/blob/master/apps/testing_interface.html`, 2019.

[8] S. Acquaviva, Y. Pu, M. Kryven, T. Sechopoulos, C. Wong, G. Ecanow, M. Nye, M. Tessler, and J. B. Tenenbaum, "Communicating Natural Programs to Humans and Machines," in *NeurIPS*, 2022.

[9] Lab42, "ARCreate." `https://lab42.global/arcreate/`, 2024.

[10] NestJS, "[NestJS] Documentation." `https://nestjs.com/`, 2024. Accessed: 2024-02-19.

[11] H. Lee, S. Kim, S. Lee, S. Hwang, J. Lee, B.-J. Lee, and S. Kim, "ARCLE: The Abstraction and Reasoning Corpus Learning Environment for Reinforcement Learning," in *CoLLAs*, 2024.

Appendix A

Abbreviations

AGI Artificial General Intelligence

ARC Abstraction and Reasoning Corpus

O2ARC Object-Oriented ARC

ARCLE ARC Learning Environment