ALLURE*: A Multi-modal Guided Environment for Helping Children Learn to Solve a Rubik's Cube with Automatic Solving and Interactive Explanations

Kausik Lakkaraju[1], Thahimum Hassan[2], Vedant Khandelwal[1], Prathamjeet Singh[2], Cassidy Bradley[2], Ronak Shah[1], Forest Agostinelli[1], Biplav Srivastava[1], Dezhi Wu[2]

[1] AI Institute, University of South Carolina, Columbia, USA

[2]Department of Integrated Information Technology, University of South Carolina, Columbia, USA

Abstract

Modern artificial intelligence (AI) methods have been used to solve problems that many humans struggle to solve. This opens up new opportunities for knowledge discovery and education. We demonstrate ALLURE, a collaborative educational AI system for learning to solve the Rubik's cube that is designed to help students improve their problem solving skills. ALLURE can both find its own strategies for solving the Rubik's cube and explain those strategies to humans. In the future, ALLURE will also be able to collaborate with humans by building on user-provided strategies for solving the Rubik's cube. Interaction between AI and user is facilitated by visual and natural language modalities.

Keywords: Rubik's Cube, Education, Reinforcement learning, Explainable AI, Chatbots, Human Computer Interfaces

Introduction

We seek to create a platform for knowledge discovery where humans can collaborate with artificial intelligence (AI) to obtain new ideas and to test their own ideas for solving a problem (Agostinelli et al. 2021). This has applications to both education as well as the frontiers of research. Our project and its prototype, ALLURE, seeks to address the gap and role that today's AI technology can play in education through a Rubik's cube use case. Our tool also contributes to a broader scope of today's education through interactive, multi-modal, and explainable AI (XAI) user interfaces to facilitate and develop students' twenty-first century skills, including communication, collaboration, problem-solving, critical, and creative thinking to foster lifelong learning and digital citizenship (Mayer and Wittrock 2006; Ng 2015; Conn and McLean 2019).

We pick the Rubik's cube given its apparent ability to capture the imagination of people around the world (Rubik 2020) as well as its relationship to mathematics (Joyner 2008) and computer science (Korf 1997; Rokicki et al. 2014; Agostinelli et al. 2019). Its ability to foster problem-solving abilities can be seen by international "cubing" competitions, including competitions for solving the cube in the fastest time, solving the cube in the in the fewest number of moves, and solving multiple cubes blindfolded.

*Name derived from the underlined letters.

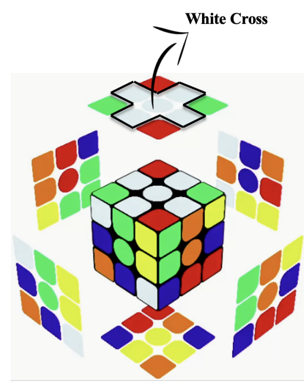

Figure 1: The Solved white cross

In this demonstration, we will show how ALLURE is able to successfully find a strategy for achieving the white cross (shown in Figure 1) and use both a visual and natural language interface to explain its strategy to the user. A step-by-step illustration of the visual interface is shown in Figure 2.

Explainable AI Methods

Solving puzzles such as the Rubik's cube has been of interest to the computer science community for decades (Korf 1997; Rokicki et al. 2014). Recently, a method called DeepCubeA (McAleer et al. 2018; Agostinelli et al. 2019) used AI techniques to create a domain-independent algorithm that learns to solve the Rubik's cube, among other puzzles. This automated problem-solving algorithm has since been applied to cryptography (Jin and Kim 2020) and quantum computing (Zhang et al. 2020). However, like previous methods, DeepCubeA cannot explain to a human its overall strategy behind its solutions and cannot build on human-provided strategies.

To give DeepCubeA the ability to produce explainable solutions, we draw on inspiration from how humans explain the many different methods for solving the Rubik's cube (Aggarwal 2017). In almost all of these methods, there is a series of subgoals to be achieved and each subgoal has a set of algorithms used to achieve it. Therefore, ALLURE will collaborate with the user by communicating subgoals and algorithms it has discovered while building on subgoals and

(a) (b) (c) (d) (e)

Figure 2: Solving *White Cross* problem in Rubik's cube. (a) Initial state: *White-Orange Edge Piece* is not in place, (b) Step 01: perform D', (c) Step 02: perform F', (d) Step 03: perform: R, (e) Goal state: perform F to get *White Cross*.

algorithms suggested by the user.

To allow DeepCubeA to quickly adapt to new subgoals, we use hindsight experience replay (Andrychowicz et al. 2017) to train a version of DeepCubeA that can achieve any valid Rubik's cube configuration, including subgoals where only a subset of the stickers are specified. DeepCubeA can then be used to provide us with examples of how to achieve any particular subgoal. For example, if we would like to achieve subgoal i, we first scramble the cube by taking hundreds of random moves, then use DeepCubeA achieve subgoal $i-1$, then collect examples of the states seen and actions taken when going from subgoal $i-1$ to subgoal i.

From these examples, we now need to extract human-understandable algorithms. To accomplish this, we use first-order inductive logic programming (ILP) (Muggleton 1992). ILP gives us the ability to define relational predicates that take advantage of symmetry and, using the appropriate background knowledge, can describe algorithms in a concise logic program. We use the Popper ILP software (Cropper and Morel 2021) that also allows us to include expressive inductive biases to simplify the search for a program. For example, we can give the program common sense rules, such as a specifying that a sticker cannot be in more than one location at a time. We can also use ILP to discover a hierarchy of subgoals by finding patterns in the order in which algorithms are applied and the learned preconditions for those algorithms. We now turn to ILP and HCI methods for the collaboration between human and AI.

Collaborative Interface

User interface (UI) design is challenging to effectively display solutions to the Rubik's Cube with understandable problem-solving steps to demo and engage users with real-time cube rotations driven by AI algorithms. Moreover, allowing the user to communicate their own ideas adds another layer of difficulty for such a design. To facilitate the initial engagement and active learning for this cube problem-solving, we design and implement (1) a series of multi-modal user interfaces through 3D model visualizations to visually present Rubik's Cube and real-time cube manipulations with explainable steps obtained from the AI system (this portion builds on code from (cfop and Hui 2020) and visualizations from (YouCanDotheRubik'sCube 2021)) (2) an interactive chatbot system to provide learners customized pedagogical hints and guidance, afforded by multi-modal UIs via text, image, animation and voice.

We use NLP techniques while having the collaborative assistant (chatbot) interact with the user to understand their gaming and learning intents, track progress and manage interactions. The user can refer to previous conversation as part of their context to initiate a game. The system will be able to track user's conversation and assess emotions like sentiments to detect frustration, user engagedness and learning progress. The system translates explanations from first order logic to natural text using templates that are customizable to different emotion states of the user. We use Rasa (Bocklisch et al. 2017) to implement this NLP interface. However, we recognize the need for careful user evaluation to determine the effectiveness of these methods for different user groups given their susceptibility to trust issues like bias (Srivastava et al. 2020; Kiritchenko and Mohammad 2018). Furthermore, users will have the ability to communicate their own ideas for subgoals using a virtual cube that they can edit. Users will be able to communicate their ideas for algorithms by providing the system with examples of the algorithm and having the ILP system induce a logic program, which will then be translated to natural language for verification. Alternatively, the user can directly communicate their algorithm using natural language, with the system asking for clarification when encountering any ambiguities.

Discussion and Conclusion

In the demonstration, we first visually introduce the user to the Rubik's cube and conventions of moves. Then, an AI agent (chatbot) explains how it can solve a problem and will explain the steps. It learns the *cross* and explains the solution with visual cues. The agent gives a natural language explanation of the algorithm being used and its effects. Next, a step-by-step visualization shows the moves required to obtain the cross.

To our knowledge, ALLURE is the first tool to both learn to solve Rubik's cube and explain its solutions to humans. In the future, we will incorporate collaborative learning, personalized interventions, multi-lingual capabilities, and conduct user studies with students.

References

Aggarwal, A. 2017. List of methods. https://www.speedsolving.com/wiki/index.php/List_of_methods.

Agostinelli, F.; Mavalankar, M.; Khandelwal, V.; Tang, H.; Wu, D.; Berry, B.; Srivastava, B.; Sheth, A.; and Irvin, M. 2021. *Designing Children's New Learning Partner: Collaborative Artificial Intelligence for Learning to Solve the Rubik's Cube*, 610–614. New York, NY, USA: Association for Computing Machinery. ISBN 9781450384520. URL https://doi.org/10.1145/3459990.3465175.

Agostinelli, F.; McAleer, S.; Shmakov, A.; and Baldi, P. 2019. Solving the Rubik's cube with deep reinforcement learning and search. *Nature Machine Intelligence* 1(8): 356–363.

Andrychowicz, M.; Wolski, F.; Ray, A.; Schneider, J.; Fong, R.; Welinder, P.; McGrew, B.; Tobin, J.; Abbeel, O. P.; and Zaremba, W. 2017. Hindsight experience replay. In *Advances in Neural Information Processing Systems*, 5048–5058.

Bocklisch, T.; Faulkner, J.; Pawlowski, N.; and Nichol, A. 2017. Rasa: Open source language understanding and dialogue management. *arXiv preprint arXiv:1712.05181* .

cfop; and Hui, H. G. 2020. Three-rubiks-cube. https://github.com/lab89/three-rubiks-cube.

Conn, C.; and McLean, R. 2019. *Bulletproof Problem Solving: The One Skill that Changes Everything*. John Wiley & Sons.

Cropper, A.; and Morel, R. 2021. Learning programs by learning from failures. *Machine Learning* 110(4): 801–856.

Jin, J.; and Kim, K. 2020. 3D CUBE Algorithm for the Key Generation Method: Applying Deep Neural Network Learning-Based. *IEEE Access* 8: 33689–33702.

Joyner, D. 2008. *Adventures in group theory: Rubik's Cube, Merlin's machine, and other mathematical toys*. JHU Press.

Kiritchenko, S.; and Mohammad, S. 2018. Examining Gender and Race Bias in Two Hundred Sentiment Analysis Systems. In *Proceedings of the Seventh Joint Conference on Lexical and Computational Semantics*, 43–53. New Orleans, Louisiana: Association for Computational Linguistics. doi:10.18653/v1/S18-2005. URL https://aclanthology.org/S18-2005.

Korf, R. E. 1997. Finding Optimal Solutions to Rubik's Cube Using Pattern Databases. In *Proceedings of the Fourteenth National Conference on Artificial Intelligence and Ninth Conference on Innovative Applications of Artificial Intelligence*, AAAI'97/IAAI'97, 700–705. AAAI Press. ISBN 0-262-51095-2. URL http://dl.acm.org/citation.cfm?id=1867406.1867515.

Mayer, R. E.; and Wittrock, M. C. 2006. Problem solving. *Handbook of educational psychology* 2: 287–303.

McAleer, S.; Agostinelli, F.; Shmakov, A.; and Baldi, P. 2018. Solving the Rubik's Cube with Approximate Policy Iteration. In *International Conference on Learning Representations*.

Muggleton, S. 1992. *Inductive logic programming*. 38. Morgan Kaufmann.

Ng, W. 2015. *New Digital Technology in Education*. Springer. doi:10.1007/978-3-319-05822-1_1.

Rokicki, T.; Kociemba, H.; Davidson, M.; and Dethridge, J. 2014. The diameter of the Rubik's Cube group is twenty. *SIAM Review* 56(4): 645–670.

Rubik, E. 2020. *Cubed: The Puzzle of Us All*. Flatiron Books. ISBN 1250217776.

Srivastava, B.; Rossi, F.; Usmani, S.; and Bernagozzi, M. 2020. Personalized Chatbot Trustworthiness Ratings. *IEEE Transactions on Technology and Society* 1(4): 184–192. doi:10.1109/TTS.2020.3023919.

YouCanDotheRubik'sCube. 2021. Rubik's Cube (3x3) Online Solution. https://www.youcandothecube.com/solve-it/3x3-solution.

Zhang, Y.-H.; Zheng, P.-L.; Zhang, Y.; and Deng, D.-L. 2020. Topological Quantum Compiling with Reinforcement Learning. *Physical Review Letters* 125(17): 170501.