(A)I Will Teach You to Play Gomoku: Exploring the Use of Game AI to Teach People

Qiao Zhang
qiao.zhang@drexel.edu
Drexel University
Philadelphia, Pennsylvania, USA

Christopher J. MacLellan
christopher.maclellan@drexel.edu
Drexel University
Philadelphia, Pennsylvania, USA

ABSTRACT

Artificial intelligence systems such as AlphaGo, AlphaGo Zero and AlphaZero, have demonstrated their advantages and competency over human players. However, little research has explored the possibility of applying such algorithms for educational purposes, such as teaching people to play strategy games. To investigate this gap, we designed and developed a Gomoku tutor that can provide instant/delayed feedback to users. We trained an expert model for Gomoku from scratch by using an open-source AlphaZero implementation and embedded this model into our Gomoku tutoring system. We plan to use this tutor to investigate two main research questions: 1) Can Game AI models, which are inhuman in their expertise, provide guidance that improves human learning? 2) How do different types of Game AI derived feedback affect people's learning outcomes? In this paper, we outline our experimental plans to investigate these questions.

CCS CONCEPTS

• **Human-centered computing → HCI design and evaluation methods**; **User studies**.

KEYWORDS

Gomoku, intelligent tutor, strategic decision making

ACM Reference Format:
Qiao Zhang and Christopher J. MacLellan. 2022. (A)I Will Teach You to Play Gomoku: Exploring the Use of Game AI to Teach People. In *Proceedings of the Ninth ACM Conference on Learning @ Scale (L@S '22), June 1–3, 2022, New York City, NY, USA.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3491140.3528331

1 INTRODUCTION

Go is an ancient strategy game that is simple to play, but difficult to master. Although early Artificial Intelligence (AI) work demonstrated that computational systems can beat human experts at Chess [2], many believed AI systems would never be able to master Go because of its substantially larger problem space. Despite this early skepticism, emerging Game AI approaches, such as AlphaGo [5], AlphaGo Zero [7] and AlphaZero [6], have successfully defeated human Go grand masters—demonstrating definitively that computational mastery of Go is possible. Building on these emerging advancements, this work investigates the use of Game AI models to support human learning. Specifically, we aim to explore the idea that Game AI expert models, such as those generated by AlphaZero, might be used to power intelligent tutoring systems that can teach human students to play strategy games, such as Go. Ultimately, we envision a future where Game AI models can be leveraged to transform human novices into a grand masters.

When exploring this concept, we have encountered many open questions. First, the expertise exhibited by Game AI systems is qualitatively very different from that exhibited by human experts. For example, the AlphaZero system uses an approach called Monte Carlo Tree Search to simulate tens of thousands of moves to support each decision. In contrast, it is widely believed that human experts only consider tens of moves to support each decision. Additionally, expertise in AlphaZero is represented as statistical weights on a neural network, so decision making for this systems is difficult to explain, whereas humans can explain and justify their decision making. Given the fundamental differences between humans and Game AI expertise, it is unclear whether Game AI models will be compatible with how humans think and reason. Will humans be able to learn from tutors powered by these models? Second, it is unclear which kinds of feedback and guidance we should use Game AI models to provide. Typically, intelligent tutoring systems provide students with a combination of contextualized on-demand hints and immediate correctness feedback [10]. However, it is unclear if this kind of support is appropriate or effective for teaching Go knowledge.

To better understand how Go is typically taught and learned, we joined a local Go club, attended weekly online meetings for three months, and met with multiple Go experts. From these experiences, we learned that the standard practice for teaching novices is to have them play many games and to engage in after-action reviews of these games. During after-action reviews, Go teachers would highlight key good or bad moves and discuss possible alternative moves with novice learners. Surprisingly, we did not observe Go teachers providing novices with on demand hints or immediate feedback on their moves while playing the game. These observations produced many questions we hope to investigate in this work. For example, are common tutoring practices, such as hints and immediate feedback, effective for teaching Go knowledge? Or, is it better to provide learners with delayed feedback in the form of after-actions reviews?

Although, our long-term goal is to explore the construction of an intelligent Go tutor, this initial work investigates these questions in the context of Gomoku, a game similar to tic-tac-toe that is played

L@S '22, June 1–3, 2022, New York City, NY, USA

ACM ISBN 978-1-4503-9158-0/22/06.
https://doi.org/10.1145/3491140.3528331

on a Go board where players win by getting five marks in a row. In our preliminary investigations, we determined that it takes novices many years of dedicated study to reach mastery of Go. In contrast, Gomoku is something that can be mastered more quickly over the course of an experimental study. Gomoku shares some similarities with Go, but is simpler making it easier to use to investigate our questions.

2 BACKGROUND

Intelligent tutoring systems aim to support human learners by emulating the kinds of instructional support that human tutors provide. Typically tutoring systems are characterized as consisting of two algorithmic loops: an outer and an inner loop [10]. The outer loop intelligently selects which problems a learner should work on next, to focus their practice on previously unmastered skills. The inner loop provides contextualized hints and immediate correctness feedback. To power these loops, a tutoring system leverages an expert model, which is a model of correct and incorrect behavior on the target task.

Building an expert model is typically viewed as one of the most challenging parts of building an intelligent tutoring system [1]—requiring substantial time and technical expertise. While building an expert models for educational tasks, such as Mathematics tasks, is difficult, building an expert model for a game like Chess, Go, or Gomoku is much more challenging. Unlike math tutors, these games have substantially large solution spaces, a characteristic that is known to make expert model authoring more difficult [3].

Fortunately, emerging Game AI systems, such as AlphaZero [6], specifically address the problem of authoring an expert model for these Games. These systems use a combination of deep learning and Monte Carlo tree search to acquire an expert model through self play (playing the game against copies of itself). Using this approach, these systems can learn an expert model capable of beating a Go grand master. To support our work, we utilized an open-source implementation of the AlphaZero algorithm that was designed specifically for Gomoku [8] (also called Gobang or Five in a Row). This implementation inspired us to examine the feasibility of designing and building a Go/Gomoku tutor with pre-trained models to provide instant or delayed feedback to users. AlphaZero expert models can not only serve as an AI opponent, but can also make recommendations about the next best moves and generate feedback on the quality of a human players move.

Training an expert model for Go takes substantial compute. Fortunately, Gomoku is simpler than Go or chess, with a smaller state space. As a result, training an expert model for Gomoku using AlphaZero is more manageable and cost effective than training an expert model for Go. We plan to conduct our human-subjects experiments using a Gomoku expert model, but we believe our approach could be transferred into the process of learning and tutoring more complex strategy games such as Go.

3 RESEARCH QUESTIONS

We have two main research questions to address through this study.

RQ1: Can Game AI models, which are inhuman in their expertise, provide guidance that improves human learning?

(1) Can a reinforcement learning model teach people to make better strategic decisions?
(2) Is it possible for a machine to teach people and provide a learning experience that is at least as good as a human expert?

RQ2: How do different types of Game AI derived feedback affect people's learning?

(1) Do learners benefit more from on demand hints and immediate feedback or hints and feedback provided during an after action review?
(2) Does providing players with feedback and hints improve their learning over simply playing games and getting win/loss feedback?

4 EXPERIMENTAL DESIGN

We developed a Gomoku tutor and experimental design to test the following three hypotheses related to our research questions:

- Models of expertise acquired by deep reinforcement learning can be used within tutoring systems to provide people with feedback on which decision is best at each point within a strategic task where decision feedback is typically delayed (i.e., where you do not get feedback on whether a move was good until the end of the game based on whether you win or lose).
- People that received feedback and hints will have better learning than those that do not.
- When people receive immediate feedback (rather than a delayed version of the same feedback), they will have better learning efficiency.

We plan to recruit 90 people from Amazon Mechanical Turk (MTurk) and randomly assign them into three groups with 30 people in one group. Each participant will interact with our Gomoku tutor for a 15-minutes training session. We will have two experimental groups and one control group. In the two experimental groups, instant or delayed feedback will be provided to the participants on each move they make. The participants can decide whether they want to use this information to perform further steps.

Table 1 describes the condition of each group.

- **Group1** (delayed, outcome feedback): Participants assigned into this group will not receive any feedback from the tutor. They will only be told whether they won or loss each game.
- **Group2** (delayed, action feedback): Participants assigned to this group will receive feedback on the quality of each move and will be able to request hints regarding the best move in a given state. However, these hints and feedback will be delayed until an after-action review the learner can engage in after the game is ended.
- **Group3** (instant, action feedback): Participants assigned into this group will receive instant feedback and hints from the tutor while playing the game.

To better address our research questions and test our hypothesis, we designed a survey to collect quantitative and qualitative data from the participants at the end of the experiment. Below are some sample statements we included in a survey. The participants are

Table 1: Description of Each Study Group

	Instant	Delayed
Binary Outcome (win/lose outcome, no hints/feedback)	N/A	Group1 (baseline)
AI-Generated Help (Instant feedback on moves and best next step hints)	Group3	Group2

asked to rate the statements on a scale of 1 to 5, with 1 being strongly disagree and 5 being strongly agree.

- I know how to play Gomoku before attending this experiment.
- I learned how to play Gomoku from this experiment.
- I think Gomoku is difficult to learn.
- Practicing with the Gomoku training system improved my learning.
- The feedback from the Gomoku training system helped me to learn.
- I understood the feedback from the Gomoku training system.
- I ignored the feedback from the Gomoku training system.
- I found the feedback from the Gomoku training system distracting.
- The Gomoku AI made mistakes when playing the next steps.
- The Gomoku AI made mistakes when giving me hints.

Our study, including the use of data to be collect from the experiment, was reviewed and approved by Drexel's Institutional Review Board.

5 GOMOKU TUTOR

We designed and developed the Gomoku tutor using Flask and Python with a SQLite database. Participants go through five stages when working with our Gomoku Tutor: Consent, Instructions, Training, Testing and Survey. During the Consent stage, we introduce who we are, the purpose of our study, the approximate time of this experiment, the upcoming stages, the kinds of data we are going to collect, how the data will be saved and used, and the compensation they will receive by the end of the experiment.

In the Instructions stage, we verbally describe the rule of Gomoku and how to use the tutor. Figure 1 displays the instructions we present to participants.

We intend to have three conditions in the Training stage (baseline, instant and delayed). We have implemented the baseline condition and the instant condition by the time of this paper. Participants always play black and go first. The training time is 15 minutes, during which participants will learn how to play Gomoku by interacting with the tutor. They will receive instant or no feedback depending on which condition they entered. During the training stage, participants can end a game and start a new game at any time regardless of the board state. Figure 2 demonstrates our interface design for participants in the instant condition. By clicking on the "Please Give me a Hint" button, users are presented with a best next move hint represented by a in dashed circle on the board. They have the option to take the next move on this suggested location or on another location of their choosing. Whenever a player makes

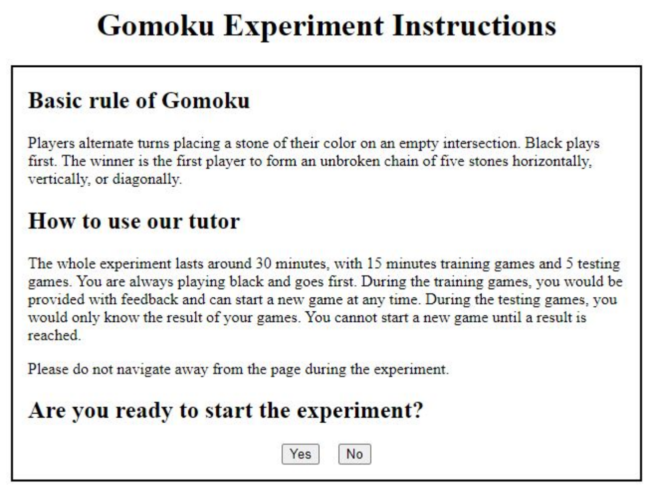

Figure 1: The Instructions stage of our Gomoku tutor.

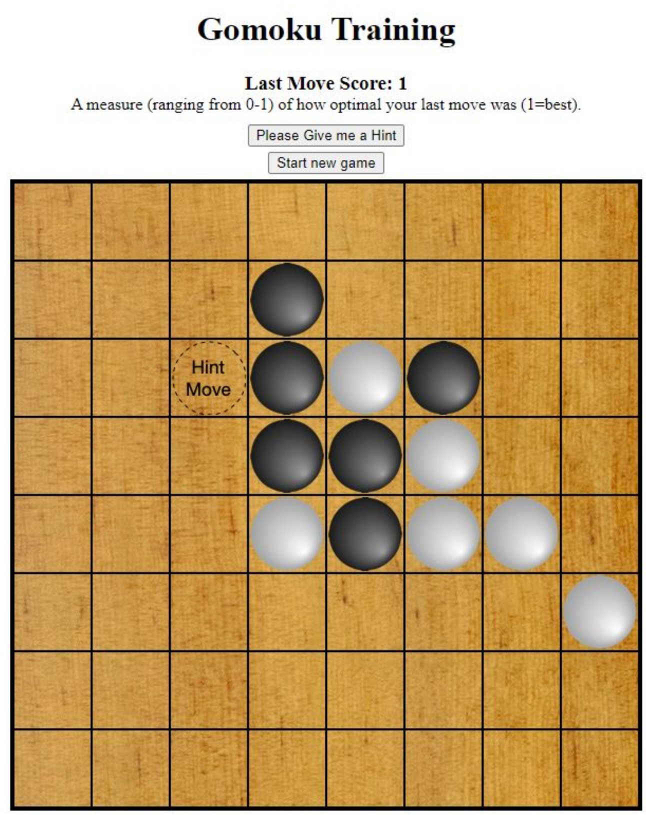

Figure 2: The layout of instant condition (Group3) with suggested hint move in Training stage. Our tutor is demonstrating a strategy of making two open-ends with the hint move.

a move, they receive feedback on their move in the form of a score ranging from 0 to 1 (0=worst and 1=best). The hinted action is always close to 1. In the baseline condition, participants can only see and interact with the "Start new game" button without getting instant feedback on each step they played. In the delayed condition, participants have the ability to engage in an after action review. During the after action review, participants can step forward and backward through the moves of the game and see the same move scores and hints that participants in the instant feedback condition receive.

When training ends, participants will play 5 testing games with no feedback. They will only be informed about whether they win/lose at the end of each game. There's no time limit on making a move. According to our internal testing results, it takes around 10 to 15 minutes for a participant to complete all testing games. To evaluate participants' learning outcomes and test our hypothesis, we use five different AlphaZero models trained for different amounts of time and with different hyper-parameters. We selected these models so that the evaluation games increase in difficulty from the first game to the last.

Our experiment ends with a survey that collects quantitative and qualitative data from the participants. Upon completing the survey, the participants receive a code they enter into their MTurk account to get paid.

6 GOMOKU EXPERT MODEL

We trained multiple strategy models using an open-source implementation of AlphaZero algorithm specifically designed for Gomoku [8] from scratch. We trained our model in self play on a server with an A40 GPU for around two weeks. The model we used during internal testing has the longest training time with 23500 epochs. We tested our model with 7 participants in a pilot study and the learned model beat or tied with all the participants. If participants in the instant feedback condition, request hints for every step, they will tie with the model in the end.

However, when we evaluated the policy models by having each model play with another model for 10 sets of games and ranked them by wining rate, we noticed that the model competency is not always consistent, indicating that the model with the longest training time does not necessarily always win. This makes it difficult to identify the best model to provide feedback to participants; however, for our work we just use the model that has been trained the longest. Since Gomoku is a simpler game than Go, we believe that if we train the model for long enough we should be able to obtain an *optimal* expert model.

7 RELATED WORKS

The work closest to ours is a study conducted by John Stamper and Steven Moore [4, 9] where they explored the problem solving strategies of both humans and AI agents in the open-ended domain of video games. They demonstrated how closely the agent policies resemble the real-world problem solving of a human player and explored the possibility of extracting human-level strategies for agent policies. However, they did not explore the use of Game AI expert models for tutoring. It would be interesting to explore the outcome of using AI strategies to provide feedback to people in completing strategic tasks assuming they have similar expertise as human experts.

8 CONCLUSIONS AND FUTURE WORK

Our next step is to implement the delayed feedback condition for our proposed Gomoku experiment, launch the experiment on Amazon's Mechanical Turk, and recruit participants. Upon collecting the data, we will investigate the participants' learning within each group, identify if there is any improvement in their Gomoku skill as a result of training in different conditions. Finally, we will analyze the performance across groups and see if there is any significant difference between the instant feedback condition (Group3) and the delayed feedback condition (Group2).

Moving forward, we may make minor modifications to the tutor design by adding a *Pretest* stage after the Instructions stage. In this way, we can conduct pre-posttest analysis to better compare participants' learning outcomes across groups and evaluate the effectiveness of our tutor with different levels of assistance. We may also increase the training time and the number of testing games as we launch the experiment.

ACKNOWLEDGMENTS

This work was funded in part by the DARPA POCUS program (award #HR00112190076), the NSF AI-ALOE institute (award #2112532), and the ARL STRONG program (award #W911NF2120101). The views, opinions and/or findings expressed are those of the author and should not be interpreted as representing the official views or policies of these funding agencies. We would like to thank Dr. Denise E. Agosto for providing feedback on early-stage experimental design. We thank Adit Gupta, Glen Smith, Natasha Lalwani, Zeyu Chen, and Harshil Thakur from the Teachable AI Lab for participating in internal testing and providing feedback on early version of the tutor.

REFERENCES

[1] Vincent Aleven, Bruce M McLaren, Jonathan Sewall, Martin Van Velsen, Octav Popescu, Sandra Demi, Michael Ringenberg, and Kenneth R Koedinger. 2016. Example-tracing tutors: Intelligent tutor development for non-programmers. *International Journal of Artificial Intelligence in Education* 26, 1 (2016), 224–269.

[2] Murray Campbell, A Joseph Hoane Jr, and Feng-hsiung Hsu. 2002. Deep blue. *Artificial intelligence* 134, 1-2 (2002), 57–83.

[3] Christopher J MacLellan, Erik Harpstead, Eliane Stampfer Wiese, Mengfan Zou, Noboru Matsuda, Vincent Aleven, and Kenneth R Koedinger. 2015. Authoring Tutors with Complex Solutions: A Comparative Analysis of Example Tracing and SimStudent.. In *Workshops at the 17th International Conference on Artificial Intelligence in Education AIED 2015*.

[4] Steven Moore and John C Stamper. 2019. Exploring Expertise through Visualizing Agent Policies and Human Strategies in Open-Ended Games.. In *EDM (Workshops)*. 30–37.

[5] David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. 2016. Mastering the game of Go with deep neural networks and tree search. *nature* 529, 7587 (2016), 484–489.

[6] David Silver, Thomas Hubert, Julian Schrittwieser, Ioannis Antonoglou, Matthew Lai, Arthur Guez, Marc Lanctot, Laurent Sifre, Dharshan Kumaran, Thore Graepel, et al. 2018. A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play. *Science* 362, 6419 (2018), 1140–1144.

[7] David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas Baker, Matthew Lai, Adrian Bolton, et al. 2017. Mastering the game of go without human knowledge. *nature* 550, 7676 (2017), 354–359.

[8] Junxiao Song. 2019. AlphaZero-Gomoku. https://github.com/junxiaosong/AlphaZero_Gomoku.

[9] John Stamper and Steven Moore. 2019. Exploring teachable humans and teachable agents: Human strategies versus agent policies and the basis of expertise. In *International Conference on Artificial Intelligence in Education*. Springer, 269–274.

[10] Kurt VanLehn. 2006. The behavior of tutoring systems. *International journal of artificial intelligence in education* 16, 3 (2006), 227–265.