Identifying Critical Pedagogical Decisions through Adversarial Deep Reinforcement Learning

Song Ju, Guojing Zhou, Hamoon Azizsoltani, Tiffany Barnes, Min Chi
Department of Computer Science
North Carolina State University
Raleigh, NC 27695
{sju2, gzhou3, hazizso, tmbarnes, mchi}@ncsu.edu

ABSTRACT

For many forms of e-learning environments, the system's behaviors can be viewed as a sequential decision process wherein, at each discrete step, the system is responsible for deciding the next system action when there are multiple ones available. Each of these system decisions affects the user's successive actions and performance and some of them are more important than others. Thus, this raises an open question: how can we identify the critical system interactive decisions that are linked to student learning from a long trajectory of decisions? In this work, we proposed and evaluated Critical-Reinforcement Learning (Critical-RL), an adversarial deep reinforcement learning (ADRL) based framework to identify critical decisions and induce compact yet effective policies. Specifically, it induces a pair of adversarial policies based upon Deep Q-Network (DQN) with opposite goals: one is to improve student learning while the other is to hinder; critical decisions are identified by comparing the two adversarial policies and using their corresponding Q-value differences; finally, a Critical policy is induced by giving optimal action on critical decisions but random yet reasonable decisions on others. We evaluated the effectiveness of Critical policy against a random yet reasonable (Random) policy. While no significant difference was found between the two condition, it is probably because of small sample sizes. Much to our surprise, we found that students often experience so-called *Critical phase*: a consecutive sequence of critical decisions with the same action. Students were further divided into High vs. Low based on the number of Critical phases they experienced and our results showed that while no significant was found between the two Low groups, the High Critical group learned significantly more than the High Random group.

Keywords

Reinforcement Learning, Critical Decision, Deep Q-Network, Adversarial Reinforcement Learning

1. INTRODUCTION

Intelligent Tutor Systems (ITSs) are a type of highly interactive e-learning environment which facilitates learning by providing contextualized feedback and step-by-step support to individual students [4, 14]. These step-by-step behaviors can be viewed as a sequential decision process where at each step the system chooses an action (e.g. give a hint, show an example) from a set of options. During tutoring, the system makes a series of decisions to provide adaptive instructions. Some of the decisions might be *more important and impactful than others*. This raises a major open question: ***How can we identify the critical system interactive decisions that are linked to student learning especially in a long trajectory of decisions?*** For example, in our ITS, the tutor makes more than 400 sequential decisions during training.

Reinforcement Learning (RL) offers one of the most promising approaches to data-driven decision-making applications and RL algorithms are designed to induce effective policies that determine the best action for an agent to take in any given situation so as to maximize some predefined cumulative reward. A number of researchers have studied the application of existing RL algorithms to improve the effectiveness of ITSs [2, 13, 12, 3]. However, relatively little work has been done to analyze, interpret, and explain RL-induced policies. While traditional hypothesis-driven, cause-and-effect approaches offer clear conceptual and causal insights that can be evaluated and interpreted, RL-induced policies are often large, cumbersome, and difficult to understand. In this work we propose to induce compact RL policies that highlight key or critical decisions by taking advantage of the structure of the domain and the structure of our induced policies by leveraging the conditional independence relationships among the state features.

We propose Critical-RL, an adversarial deep reinforcement learning (ADRL) based framework to induce compact policies that make critical decisions. By inferring critical decisions we can identify which tutor actions are minimally necessary for the tutoring process to be effective, which holds great implications for systems research. Additionally, inference of critical relationships is one of the central tasks of science and it is one of the most challenging topics in many disciplines, particularly in the areas where controlled experiments are comparatively expensive or even impossible. In this work, we propose a general framework that fully integrates automatic critical inference and standard

reinforcement learning in an ITS setting. We expect that our framework can be spread to other similar domains for related critical tasks.

2. RELATED WORK

2.1 Applying RL to ITSs

Prior work has applied a variety of RL approaches to induce pedagogical policies to improve the effectiveness ITS [1, 6, 15, 10]. For example, Beck et al. [1] applied temporal difference learning on simulated students to induce a policy that would minimize student time on task. Results showed that the policy group indeed spent significantly less time than the non-policy group. Shen et al. [13] applied MDP to induce a pedagogical policy aimed at improving students' learning performance. Results showed that the induced policy was significantly more effective than a random baseline policy for certain learners. Mandel et al. [6] applied a POMDP based approach to induce a pedagogical policy targeted at improving students' learning gain. The RL-induced policy was then compared with an expert policy and a random baseline policy. Results revealed that the RL-induced policy significantly outperformed the other two. Wang et al.[15] applied a variety of deep RL approaches on simulated students to induce pedagogical policies that would improve students' normalized learning gain in an educational game. Simulation results suggested that deep RL policies were more effective than linear model based RL policies.

To summarize, prior work has shown that RL induced policies can lead to improved student learning/behavior as compared to baseline policies. However, prior work has mainly focused on inducing effective policies from pre-collected data or simulated student, but put relatively less effort to identify the exact part that makes them effective.

2.2 Deep Reinforcement Learning (DRL)

Recent advance in deep learning has allowed RL to work in complex interactive environments which was often impractical in before. Recent work showed that RL can induce effective policies for a variety of tasks, such as game playing [8, 9], robotic control [5, 19], recommendation generation [17, 16] and also ITS control [15, 10]. However, all of the state-of-art RL algorithms focused on inducing effective policies. None of them considered interpreting, explaining and identifying critical decisions from RL induced policies.

2.3 Exploiting Q-value Difference

Some prior work has exploited the Q-value difference between actions to simplify the decision-making process/problem. For example, Mitchell et al. [7] relied on the Q-value difference to select features for RL. They proposed a Q-value difference based policy evaluation metric, which was then used to guide feature selection for RL. Zhou et al. [18] relied on Q-value difference to reduce the policy space. More specifically, they applied weighted decision tree with post-pruning to extract a compact set of 529 rules from a full set of 3706 rules. During the extraction, each rule was weighted by the Q-value difference between two alternative actions and thus increased the carry-out likelihood of more important decisions. Results showed that the full RL policy and the compact DT policy together were significantly more effective than a random policy and there is no significantly difference between the full RL policy and the compact DT policy.

In sum, prior studies have used Q-value difference to measure action importance and results suggest that it is an effective measure. However, prior work used Q-value difference to reduce the feature space or the policy space, but we used it to reduce the decision space.

3. METHOD

3.1 Adversarial Reinforcement Learning

Adversarial Reinforcement Learning (ARL) is a category of RL which can induce a pair of policies for opposite goals. In our application, an *Original Policy* was induced using the original rewards and an *Inversed Policy* was induced using the inversed rewards, which is the negative value of the original rewards. We expect these two policies to have opposite goals, one to help student learn while the other to hinder them learn.

3.2 Deep Q-Network (DQN)

Deep Q-Network (DQN) is a RL algorithm that uses a deep neural network to approximate the Q-value function. The neural network takes a state as input, which is represented as a numerical vector, and outputs its estimation of the Q values for all possible actions. During training, the neural network is updated recursively following the Bellman equation shown below until converge.

$$Q_{i+1}(s,a) = \mathbb{E}_{s' \sim \varepsilon}[r + \gamma \max_{a'} Q_i(s',a')|s,a] \qquad (1)$$

where γ is a discount factor, ε is the environment and Q_i is the action-value function at the ith iteration. DQN is a *model free* approach that it is focused on estimating the action value functions from the interactions with the environment without constructing a model of the environment. Also, it is an off-policy approach that the new policy is induced based upon the historical data generated by an alternative behavior policy.

3.3 Identifying Critical Decision

Once the adversarial policies are induced, critical decisions are identified following two rules: 1) given the state, the two policies make opposite decisions and 2) the decision is important for both policies.

For a given state, rule one is tested first. If the two policies make the same decisions, it is not critical. Otherwise, rule two is tested. In order to measure the importance of the decision for each policy, we calculate the absolute Q-value difference between the two alternative actions: $\Delta Q^*(s) = |Q^*(s,a_1) - Q^*(s,a_2)|$. If this difference is greater than a threshold, the decision is considered important for the corresponding policy. In this paper ,we set the threshold to be the median Q-value difference for all decisions in our training data set.

3.4 Critical Policy Induction

In tutoring, our ITS provides students with the same 12 problems in the same order. Among them, the first and the

eighth problems are fixed to be problem solving where the students is required to solve all the steps. For the rest 10 problems, the policies decide whether to elicit the next step from the student or to directly show the student how to solve the next step.

Thus, we induced 10 pairs of adversarial policies, one for each problem. Each pair of the adversarial policies consist of two policies: an original policy and an inversed policy. The original policy was induced using the original rewards while the inversed policy was induced using inversed rewards. Other than the rewards, all other parts of the data were identical, such as state representation and transition samples.

In order to find the best policy, for each problem, we implemented two different types of neural network: Recurrent Neural Network (RNN) and Long Short Term Memory (LSTM) to induce the adversarial policies. The policies were then evaluated using Per decision importance sampling (PDIS)[11] and the better one was selected. Once the adversarial policies were induced, whether a decision was critical or not during tutoring was determined following the two rules mentioned in section 3.3.

Finally, the Critical policy is carried out partially in that if a decision is critical, it will be carried out; otherwise, the decision will be made randomly. More specifically, for a given state, the adversarial policies are queried to determine whether the decision is critical. If it is, the decision made by the original policy will be taken; otherwise, a random decision will be taken.

4. EXPERIMENT SETUP

In order to evaluate the critical-RL induced policy, we conducted a classroom study comparing the Critical policy with the Random policy. The participants of this study were undergraduate students enrolled in the Discrete Mathematics class at the Department of Computer Science at NC State University in 2018 Fall. In this study, all students were required to complete 4 phases: 1) pre-training, 2) pre-test, 3) training on Pyrenees tutor, and 4) post-test. Pyrenees tutor is a web-based ITS for probability, which covers 10 major principles of probability such as the Complement Theorem and Bayes' Rule. During the experiment, all students in both two conditions studied the same materials, received the same questions in pre-test, trained on the same tutor, examined the same questions in post-test. The only difference was the policies used in the tutor.

In this study, 120 students were randomly assigned to the Critical condition and the Random condition. Due to preparation for final exams and the length of the study, 96 student completed the study. 3 students performed perfectly in the pre-test were excluded from our subsequent statistical analysis. The final group sizes were: $N = 50$ (Critical) and $N = 43$ (Random). We performed a Chi-square test of the relationship between students' condition and their completion rate and found no significant difference between the conditions: $\chi^2(1) = 2.55$, $p = 0.11$.

5. RESULTS

Table 1 shows the mean and standard deviation (SD) of the post-test score, learning gain (LG) and total training time for the Crucial and Random condition. Contrast comparison analysis showed no significant difference between the two conditions on all three measures. Please note that although the Critical condition appeared to outperform the Random condition on learning gain (0.05 vs. 0.03), such difference was not significant (p = 0.50). One of the possible reasons is that the group size was not large enough to demonstrate significance. A post hoc power analysis revealed that a total sample of 1544 students was required to detect significance at .05 on small effects (d=.14), with 80% power using a contrast.

Table 1: Critical vs. Random

Measure	Critical	Random	P value
Post	0.71 (0.19)	0.71 (0.20)	0.91
LG	0.05 (0.18)	0.03 (0.14)	0.50
Time	121.6 (37.7)	116.3 (30.47)	0.46

Since the Critical policy was partially carry-out where only critical decisions were made following the optimal policy, we conducted an inspection into the relation between the number of critical decisions made and student learning. Through analyzing the student-system interactive logs, we found that critical decisions were always appeared in groups and each group of consecutive critical decisions had the same action. This is aligned with existing learning theory that the learning process is a continuous process. Student can stay in the same learning state during several steps but this continuous learning state is hard to be represented by current features. In other words, in the same learning state, the agent will continuous giving same actions to the student until he moves to next learning state. So, we defined *Critical Phase* as a period of consecutive critical decisions executions with the same action according to the Critical policy.

In order to analyze the impact of critical phase, we divided students into High vs. Low groups by a median split on the number of critical phases they experienced. For the Random condition, as the execution of decisions were partially agreed with the Critical policy, we ignored the actual decision and only focused on the Critical policy's decision. Thus, we had four groups based upon their critical phase number and policies: High-Random (n=20), Low-Random (n=23), High-Critical (n=27), Low-Critical (n=23). A two-way ANOVA analysis using policies {Critical, Random} and critical phase {High, Low} as two factors and the student's learning gain as the dependent measure showed a significant interaction effect $F(1, 89) = 7.163, p = 0.009$. Subsequent contrast analysis revealed that the High-Crucial group ($M = 0.098, SD = 0.2$) significantly outperformed High-Random group ($M = -0.011, SD = 0.16$): $t(89) = 2.360, p = 0.02$. However, such difference was not significant between the Low-Crucial group ($M = 0.001, SD = 0.13$) and Low-Random group ($M = 0.067, SD = 0.12$) : $t(89) = 1.42, p = 0.16$.

In terms of time on task, a two-way ANOVA analysis on condition and critical phase number showed a main effect on critical phase number: $F(1, 89) = 5.579, p = 0.020$ in

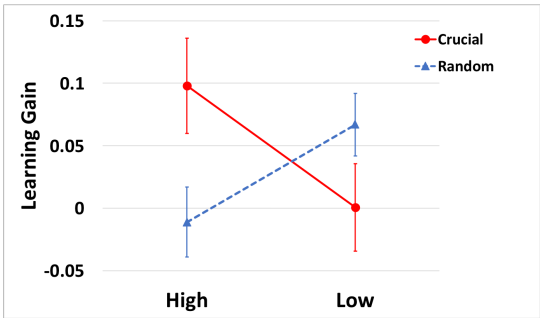

Figure 1: Comparison of Learning Performance

that the High group ($M = 127.51, SD = 36.34$) spent significantly more time on task than the Low group ($M = 110.56, SD = 30.51$). The results suggest that students experienced more critical phases are more likely to spend more time on task.

6. CONCLUSION

In this study, we proposed Critical-RL to identify critical pedagogical decisions in an ITS. Based on the ADRL framework, we induced a Critical policy which gives optimal action on critical decision points but randomly select actions on others. We empirically compared the Critical policy with a baseline Random policy in a classroom study for real students. Although there's no significant difference between the two conditions, we found the existence of Critical phase, a consecutive sequence of critical decisions with the same action. We then divided students into High vs. Low groups based on the number of Critical phases they experienced. Results showed that while the two Low groups were not sensitive to pedagogical policies, the High-Critical group significantly outperformed the High-Random group. This suggested that for certain students, the Critical policy is significantly more effective than the Random policy.

In the future, we plan to analyze the difference between states in critical and non-critical phase. Through analyzing the critical state, we hope to align critical decision with existing learning theory and further generalize our method to other domain.

7. ACKNOWLEDGMENTS

This research was supported by the NSF Grants #1432156, #1651909, #1726550, and #1916417.

8. REFERENCES

[1] J. Beck, B. P. Woolf, and C. R. Beal. Advisor: A machine learning architecture for intelligent tutor construction. In *AAAI/IAAI*, pages 552–557, 2000.

[2] M. Chi, K. VanLehn, D. Litman, and P. Jordan. Empirically evaluating the application of reinforcement learning to the induction of effective and adaptive pedagogical strategies. *User Modeling and User-Adapted Interaction*, 21(1-2):137–180, 2011.

[3] B. Clement, P.-Y. Oudeyer, and M. Lopes. A comparison of automatic teaching strategies for heterogeneous student populations. In *EDM*, 2016.

[4] K. R. Koedinger, J. R. Anderson, W. H. Hadley, and M. A. Mark. Intelligent tutoring goes to school in the big city. 1997.

[5] S. Levine, C. Finn, T. Darrell, and P. Abbeel. End-to-end training of deep visuomotor policies. *JMLR*, (17(39)):1–40, 2016.

[6] T. Mandel, Y.-E. Liu, S. Levine, E. Brunskill, and Z. Popovic. Offline policy evaluation across representations with applications to educational games. In *AAMAS*, pages 1077–1084, 2014.

[7] C. M. Mitchell, K. E. Boyer, and J. C. Lester. Evaluating state representations for reinforcement learning of turn-taking policies in tutorial dialogue christopher. *SIGDIAL*, pages 339–343, 2013.

[8] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller. Playing atari with deep reinforcement learning. *In NIPS Deep Learning Workshop*, 2013.

[9] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, and et al. Human-level control through deep reinforcement learning. *Nature*, (518):529–533, 2015.

[10] K. Narasimhan, T. Kulkarni, and R. Barzilay. Language understanding for text-based games using deep reinforcement learning. *arXiv preprint arXiv:1506.08941*, 2015.

[11] D. Precup, R. S. Sutton, and S. Singh. Eligibility traces for off-policy policy evaluation. *ICML*, pages 759–766, 2000.

[12] A. N. Rafferty, E. Brunskill, T. L. Griffiths, and P. Shafto. Faster teaching via pomdp planning. *Cognitive science*, 40(6):1290–1332, 2016.

[13] S. Shen and M. Chi. Reinforcement learning: the sooner the better, or the later the better? In *the 2016 Conference on User Modeling Adaptation and Personalization*, pages 37–44. ACM, 2016.

[14] K. Vanlehn. The behavior of tutoring systems. *International journal of artificial intelligence in education*, 16(3):227–265, 2006.

[15] P. Wang, J. Rowe, W. Min, B. Mott, and J. Lester. Interactive narrative personalization with deep reinforcement learning. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence*, 2017.

[16] X. Zhao, L. Xia, L. Zhang, Z. Ding, D. Yin, and J. Tang. Deep reinforcement learning for page-wise recommendations. *In Proceedings of the 12th ACM Conference on Recommender Systems.*, pages 95–103, 2018.

[17] G. Zheng, F. Zhang, Z. Zheng, Y. Xiang, N. J. Yuan, X. Xie, and Z. Li. Drn: A deep reinforcement learning framework for news recommendation. *World Wide Web Conference*, pages 167–176, 2018.

[18] G. Zhou, J. Wang, C. F. Lynch, and M. Chi. Towards closing the loop: Bridging machine-induced pedagogical policies to learning theories. *EDM*, 2017.

[19] Y. Zhu, R. Mottaghi, E. Kolve, J. J. Lim, A. Gupta, L. Fei-Fei, and A. Farhadi. Target-driven visual navigation in indoor scenes using deep reinforcement learning. *ICRA*, 2017.