Exploring the Impact of Simple Explanations and Agency on Batch Deep Reinforcement Learning Induced Pedagogical Policies

Markel Sanz Ausin[0000−0002−4526−9252], Mehak Maniktala, Tiffany Barnes, and Min Chi

North Carolina State University, Raleigh NC 27695, USA
{msanzau,mmanikt,tmbarnes,mchi}@ncsu.edu

Abstract. In recent years, Reinforcement learning (RL), especially Deep RL (DRL), has shown outstanding performance in video games from Atari, Mario, to StarCraft. However, little evidence has shown that DRL can be successfully applied to real-life human-centric tasks such as education or healthcare. Different from classic game-playing where the RL goal is to make an agent smart, in human-centric tasks the ultimate RL goal is to make the *human-agent interactions* productive and fruitful. Additionally, in many real-life human-centric tasks, data can be noisy and limited. As a sub-field of RL, batch RL is designed for handling situations where data is limited yet noisy, and building simulations is challenging. In two consecutive classroom studies, we investigated applying batch DRL to the task of pedagogical policy induction for an Intelligent Tutoring System (ITS), and empirically evaluated the effectiveness of induced pedagogical policies. In Fall 2018 (F18), the DRL policy is compared against an expert-designed baseline policy and in Spring 2019 (S19), we examined the impact of *explaining* the batch DRL-induced policy with student decisions and the expert baseline policy. Our results showed that 1) while no significant difference was found between the batch RL-induced policy and the expert policy in F18, the batch RL-induced policy *with simple explanations* significantly improved students' learning performance more than the expert policy alone in S19; and 2) no significant differences were found between the student decision making and the expert policy. Overall, our results suggest that pairing simple explanations with induced RL policies can be an important and effective technique for applying RL to real-life human-centric tasks.

Keywords: Deep Reinforcement Learning · Pedagogical Policy · Explanation.

1 Introduction

In interactive learning environments such as Intelligent Tutoring Systems (ITSs) and educational games, the human-agent interactions can be viewed as a temporal sequence of steps [2, 20]. Most ITSs are *tutor-driven* in that *the tutor* decides what to do next. For example, the tutor can *elicit* the subsequent step from the student either with prompting or without (e.g., in a free form entry window where each equation is a step). When a student enters an entry on a step, the ITS records its success or failure and may give feedback (e.g. correct/incorrect markings) and/or hints (suggestions for what to do next). Alternatively, the tutor can choose to *tell* them the next step directly. Each of such decisions affects the student's successive actions and performance. *Pedagogical policies* are used for the agent (tutor) to decide what action to take next in the face of alternatives.

Reinforcement Learning (RL) offers one of the most promising approaches to data-driven decision-making for improving student learning in ITSs. 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 a cumulative reward. In this work, we use *batch RL*, an RL sub-field that deals with the inability to explore the environment. In batch RL, all the learning is done from a fixed-length dataset of samples that were obtained by interacting with the environment using some unknown behavior policy. A number of researchers have studied applying RL to improve the effectiveness of ITSs (e.g. [8, 7, 11, 21, 25, 40, 39, 48, 47]). While promising, prior work has two limitations: *communication* and *agency*.

One limitation of applying RL to ITSs is *communication*. In recent years, RL, especially Deep RL, has achieved superhuman performance in several complex games [50, 51, 56, 3]. However, different from the classic game-play situations where the ultimate goal is to make the agent effective, in human-centric tasks such as ITSs, the ultimate goal is for the agent to make the *student-system interactions* productive and fruitful. Thus, we argue it is important to communicate the agent's pedagogical decisions to students. Prior work on applying RL to ITSs primarily focused on inducing effective pedagogical policies *for the tutor to act*, but the tutor rarely explains to students *why* certain pedagogical decisions are made. As far as we know, no prior research has been done on exploring the effectiveness of explaining pedagogical policies to students. On the other hand, prior research in Self-Determination Theory (SDT) suggests that explanations could be a powerful tool to increase student engagement and autonomy in learning. For example, it was shown that explaining the benefits of learning a specific task to students would increase their sense of control over their own learning [9, 44, 22, 19, 58, 49], which can improve their learning outcomes.

The other limitation of RL in ITSs is *agency*. Rather than inducing effective pedagogical policies for the tutor to act, would it be more effective if we just let students make certain pedagogical decisions? Prior research has shown that it is desirable for students to experience a sense of control over their own learning, which could enhance their motivation and engagement [9, 19] and improve their learning experience [44, 58]. People are more likely to persist in constructive activities, such as learning, exercising, or quitting smoking, when they are given choices and make decisions. Thus, we investigated the effectiveness of letting students make pedagogical decisions vs. the traditional tutor-driven approach.

In short, we 1) examined the impact of *simple* explanations of tutor pedagogical decisions on student learning, and 2) investigated the effectiveness of letting students be the decision makers. Through two empirical classroom studies, our results show that batch RL-induced policies could improve students' learning performance more than our expert-designed baseline policy *only if* simple explanations are present; and no significant difference was found between the student decision making and the baseline policy. In summary, our work suggest that neither letting the tutor make effective pedagogical policy alone nor letting students make decisions alone may be sufficient to improve student learning, a more effective way is to let the tutor make effective pedagogical decisions while communicating some of the decisions to students through simple explanations.

2 Background & Related Work

Prior Research in Applying RL to Pedagogical Policy Induction can be roughly divided into classic RL vs. Deep RL approaches. The latter is highly motivated by the fact that the combination of deep learning (neural networks) and novel reinforcement learning algorithms has made solving complex problems possible in the last decade. For instance, the Deep Q-Network (DQN) algorithm [30] takes advantage of convolutional neural networks to learn to play Atari games observing the pixels directly. Since then, DRL has achieved success in various complex tasks such as the games of Go [50], Chess/Shogi [51], Starcraft II [56], and robotic control [3]. One major challenge of these methods is *sample inefficiency* where RL policies need large sample sizes to learn optimal, generalizable policies. Batch RL, a sub-field of RL, aims to fix this problem by learning the optimal policy from a fixed set of a priori-known transition samples [24], thus efficiently learning from a potentially small amount of data and being able to generalize to unseen scenarios.

Prior research using classic RL approaches has applied both online and batch/offline approaches to induce pedagogical policies for ITSs. Beck et al. [6] applied temporal difference learning to induce pedagogical policies that would minimize the students' time on task. Similarly, Iglesias et al. applied Q-learning to induce policies for efficient learning [15, 16]. More recently, Rafferty et al. applied an online partially observable Markov decision process (POMDP) to induce policies for faster learning [34]. All of the models described above were evaluated via simulations or classroom studies, yielding improved student learning and/or behaviors as compared to some baseline policies. Offline or batch RL approaches, on the other hand, "take advantage of previous collected samples, and generally provide robust convergence guarantees" [45]. Thus, the success of these approaches depends heavily on the quality of the training data. One common convention for collecting an exploratory corpus is to train students on ITSs using *random yet reasonable* policies. Shen et al. applied value iteration and least square policy iteration on a pre-collected exploratory corpus to induce a pedagogical policy that improved students' learning performance [48, 47]. Chi et al. applied policy iteration to induce a pedagogical policy aimed at improving students' learning gain [7]. Mandel et al. [25] applied an offline POMDP to induce a policy which aims to improve student performance in an educational game. All the models described above were evaluated in classroom studies and were found to yield certain improved student learning or performance relative to a baseline policy. Wang et al. applied an online DRL approach to induce a policy for adaptive narrative generation in educational game using simulations [57]; the resulting DRL-induced policies were evaluated via simulations only. In this work, based on the characteristics of our task domain, we focus on batch RL with neural networks, also known as batch Deep Reinforcement Learning (batch DRL) [18, 13] and evaluate their effectiveness in classroom studies.

The Impact of Explanation on Learning: This work is highly motivated by large amount of research in Self-Determination Theory (SDT) investigating the benefit of explanations [10, 17, 42, 43, 36]. When teaching correlations to college students in a teacher training program, Jang et al. found that the students who were told the benefit of learning correlation (Explanation), were significantly more engaged than those who were not told (No-Explanation), in that the former showed more on-task attention, effort, and persistence than the latter [17]. Similarly, on a routine tedious task of letter copying, the Explanation students were significantly more engaged in the task than the No-Explanation peers who were not told [43]. Additionally, Reeve et al. compared the impact of Explanation vs. No-Explanation [36] on learning Chinese and found that the former self-reported significantly higher engagement in the task on a post-survey.

While explanations in much of the prior work above were human generated, in recent years an increasing amount of research has explored on how to automatically generate explanations. For example, Eslami et al. [12] investigated users' perspective on revealing advertisement algorithms and personal information used for generating personalized advertisements. As expected, users preferred interpretable explanations about how and why an ad was personalized to their identity. Additionally, Rago et al. [35] and Palanca et al. [32] explored using argumentation to provide explanations for recommender systems. More closely to this work, Barria-Pineda & Brusilovsky [5] and Tsai & Brusilovsky [53] explored explaining recommendations in education and social recommender systems and showed great promises. Despite these results, Kunkel et al. [23] showed human-generated explanations were rated more highly for recommendations and trustworthiness than machine-generated explanations based on item similarity. In [10], Deci et al. examined the impact of several factors on the effectiveness of explanations. As an example, they investigated two levels of controllingness: a high controlling statement would be something like "You *must* watch me solve this problem" while a low controlling counterpart sentence would be "Now you *can* watch me solve this problem". Results showed that low controlling explanations can be significantly more effective to enhance participants' engagement than high controlling ones and more importantly, the former can lead to a *positive correlation* between engagement and the desired learning outcomes. Inspired by this result, in this work our simple explanations are human-generated and to do so, we followed the low controlling principle.

Students as Decision Makers on ITS: While engaging students in decision-making within an ITS is not novel, prior research has focused on letting students dictate content by letting them decide *what problem* they wish to solve [20] but not *how* they wished to solve it. On one hand, letting students make their own decisions would allow them to experience a sense of control over their learning, which could enhance their motivation and engagement [9, 19] and further improve their learning experience [44, 58]. On the other hand, prior research has shown that students, especially low performing ones, may not always have the necessary meta-cognitive skills to make effective pedagogical decisions [1]. In that research, Aleven & Koedinger studied students' help-seeking behaviors in the Cognitive Tutor where students request help when they do not know what step to take next. Help is provided via a sequence of hints that progress from general top-level hints to bottom-out hints that tell them exactly what action to take. They found that students do not always have the necessary metacognitive skills to know when they need help. Roll et al., by contrast, examined the relationship between students' help-seeking patterns and their learning [38], and found that asking for help on challenging steps was productive while help-abusing behavior (asking for help as a way to avoid work) was correlated with poor learning.

3 Methods

In the conventional RL, an agent interacts with an environment $\mathcal{E}$ over a series of decision-making steps, which can be framed as a Markov Decision Process (MDP). At each timestep t, the agent observes $\mathcal{E}$ in state s_t; it chooses an action a_t from a discrete set of possible actions; and $\mathcal{E}$ provides a scalar reward r_t and evolves into next state s_{t+1}. The future rewards are discounted by the factor $\gamma \in (0, 1]$. The return at time-step t is defined as $R_t = \sum_{t'=t}^{T} \gamma^{t'-t} r_{t'}$, where T is the last time-step in the episode. The agent's goal is to maximize the expected discounted sum of future rewards, also known as the return, which is equivalent to finding the optimal action-value function $Q^*(s, a)$ for all states. Formally, $Q^*(s, a)$ is defined as the highest possible expected return starting from state s, taking action a, and following the optimal policy π^* thereafter. It can be calculated as $Q^*(s, a) = \max_\pi \mathbb{E}[R_t | s_t = s, a_t = a, \pi]$ and $Q^*(s, a)$ must follow the *Bellman Equation.* We follow the batch RL formulation in that we have a fixed-size dataset $\mathcal{D}$ consisting of all historical sample episodes, each formed by a sequence of state, action, reward tuples $(s_0, a_0, r_0, ..., s_T, a_T, r_T)$. We assume that the state distribution and behavior policy that were used to collect $\mathcal{D}$ are both unknown. We explored two batch DRL algorithms: *Deep Q-Network* (DQN) and *Double Deep Q-Network* (Double DQN).

DQN [30] is fundamentally a version of Q-learning which uses neural networks to approximate the true Q-values. In order to train the DQN algorithm, two neural networks with equal architectures are employed: one for calculating the Q-value of the current state and action $Q(s, a)$, and another neural network to calculate the Q-value of the next state and action $Q(s', a')$. The former is the main network and its weights are denoted $\boldsymbol{\theta}$ and the latter is the target network, and its weights are denoted $\boldsymbol{\theta}^{-}$. Equation 1 shows its corresponding *Bellman Equation.* It is trained through gradient descent to minimize the squared difference of the two sides of the equality. Online DQN uses an experience replay buffer to store the recently collected data and to uniformly sample (s, a, r, s') steps from it. When inducing our batch RL policy, the whole $\mathcal{D}$ is in the experience replay buffer.

$$Q(s,a;\boldsymbol{\theta}) = \mathop{\mathbb{E}}_{s'\sim\mathcal{E}}[r + \gamma \max_{a'} Q(s',a';\boldsymbol{\theta}^{-})] \tag{1}$$

Double-DQN was proposed by Van Hasselt et al. [54] by combining the idea behind Double Q-Learning [14] with the neural network advances of the DQN algorithm to form Double-DQN. The intuition behind it is to decouple the action selection from the action evaluation. To achieve this, the Double-DQN algorithm uses the main neural network for action selection first, and then the target network evaluates its Q-value. This trick has been proven to significantly reduce overestimations in Q-value calculations, resulting in better final policies. With this technique, the modified *Bellman Equation* becomes:

$$Q(s,a;\boldsymbol{\theta}) = \mathop{\mathbb{E}}_{s'\sim\mathcal{E}}[r + \gamma Q(s', \mathop{\mathrm{argmax}}_{a'} Q(s',a',\boldsymbol{\theta});\boldsymbol{\theta}^{-})] \tag{2}$$

Last but not least, in order to address the credit assignment problem caused by having delayed rewards in our ITS, we used the Gaussian Processes (GP) approach in [4] to estimate immediate rewards based on delayed rewards.

4 Pedagogical Decisions & Pedagogical Policy Induction

Pedagogical Decisions: When comparing the effectiveness of students' pedagogical decision-making vs. batch DRL, we strictly control the instructional content to be equivalent for all students in that our ITS gives students the same training problems and we focused on tutorial decisions that cover the same domain content: Problem-Solving (PS) versus Worked Examples (WE). In PS, students are given tasks or problems to complete either independently or with assistance of ITSs while in WE, students are given detailed solutions.

A great deal of research has investigated the impacts of WEs vs. PSs on learning. [52, 29, 27, 26, 37, 46, 31, 41]. Generally speaking, it is shown that studying WEs can significantly reduce the total time on task while keeping the learning performance comparable to doing PS [29, 27, 26]; alternating WE and PS can be *more effective* than PS only [52, 26, 37, 46, 31, 41]. Despite prior work, there is little consensus on how they should be combined effectively and thus when deciding between PS and WE, most existing ITSs always choose PS [20, 55]. Since there is no widespread consensus on how or when each alternative should be used, we apply batch DRL to derive pedagogical strategies directly from empirical data.

Training Corpus: Our training corpus consists of 786 historical student-ITS trajectory interactions over 5 different semesters, one trajectory per student. All students go through a standard pretest, training on ITS, and posttest procedure and each student spent around 2-3 hours on the ITS, completing around 20 training problems. To represent the learning environment, 142 state features from five categories were extracted. More specifically, we have 10 Autonomy features describing the amount of work done by the student; 29 Temporal features including average time per step, the total time spent, the time spent on PS, the time spent on WE, and so on; 35 Problem Solving features describing the difficulty of the problem, the number of easy and difficult problems solved, and so on; 57 Performance features including the number of incorrect steps, and the ratio of correct to incorrect rule applications and so on; 11 Hint-related features including the total number of hints requested etc. The primary goal of our RL-induced pedagogical policy is to improve student Learning Gain, measured by the difference between the posttest and the pretest scores with a range of $[-200, +200]$. Since in RL immediate rewards are often more effecient than delayed ones, here we applied Gaussian Processes (GP) [4] to infer the immediate rewards for non-terminal states from the final delayed reward (students' Learning Gain).

Policy Induction: For both DQN and Double DQN, we explored using Fully Connected (FC) vs. Long Short Term Memory (LSTM) to estimate the action-value function Q. Our FC network consists of four fully connected layers of 128 units each, with Rectified Linear Unit (ReLU) as the activation function. Our LSTM architecture consists of two layers of 100 LSTM units each with ReLU activation functions, and a fully connected layer as output. Additionally, for both FC and LSTM, for a given time t, we explored three input settings: 1) $k = 1$ that use only the last state s_t; 2) $k = 2$ that uses to use the last two states: s_{t-1} and s_t; and 3) $k = 3$ for using s_{t-2}, s_{t-1} and s_t. L2 regularization was employed to get a model that generalizes better to unseen data and avoid overfitting. We trained our models for 50,000 iterations, using a batch size of 200. To select the best pedagogical policy, we compared all of the different models (FC vs. LSTM, DQN vs. Double-DQN, k={1, 2, 3}) using Per-Decision Importance Sampling (PDIS), which is one of the most robust off-policy evaluation methods [33]. The policy with highest PDIS value was selected to be our final pedagogical policy. In this work, our final pedagogical policy was DQN with an LSTM network using $k = 3$ observations.

Simple Explanations: The design of our explanation is rather straightforward. We followed the "low-controllingness" principle described in [10]. Our explanations are *action-based* in that they focused on explaining the *benefit of taking the subsequent tutorial actions*. Our simple, action-based explanations were primarily based on the prior research on learning science and cognitive science. For example, a large amount of research showed that studying WEs can be more beneficial if it is a problem involving new level of difficulty or content [29, 28] and thus if the current problem was the first problem in a level, our action-based explanation for WE would state "The AI agent thinks you should view this problem as a Worked Example to learn how some new rules work." Our simple action-based explanation for other WE states: "The AI agent thinks you would benefit from viewing this problem as a worked example." Similarly, if the policy decided that the next problem should be a PS, then the message shown stated something like: "The AI agent thinks you should solve this problem yourself."

5 Experiment Setup

Our ITS is a graph-based logic tutor (name replaced for anonymity), which is used in the undergraduate Discrete Mathematics class at a large university. In this ITS, students must sequentially apply rules to logic statement nodes in order to derive the conclusion node and solve the problem. The tutor consists of seven levels, with three to four problems per level. Here level 1 is our pretest and level 7 is our the posttest. All students experience the exact same problems in the same way in the pretest and posttest. The pedagogical policy decides whether to represent each problem in the training levels 2-6 as a Worked Example (WE) or as a Problem Solving (PS). Our baseline policy is designed by the instructor who has more than 20 years experience on the subject, referred to as the *Expert-designed baseline policy* in the following. Based on our ITS, prior instructional experience, and prior research on WE vs. PS, our Expert Baseline policy is basically an alternative WE-PS policy with additional constraints: on each level, students must complete at least one PS and one WE.

Two studies were conducted: one in Fall 2018 and the other in Spring 2019, denoted F18 and S19 respectively. In both studies, our ITS was given as one of the regular homework assignments and students had one week to complete it.

For **F18**, 84 students were randomly assigned to the two conditions using stratified sampling based on the pretest score to ensure that the two conditions had similar prior knowledge. As a result, we have $N = 41$ students for the *DQN* condition and $N = 43$ for the *Expert* baseline condition. Here the tutor in the *DQN* condition followed the induced *DQN* policies described in Section 4 *without* explanations. Our stratified sampling resulted in balanced incoming competence in that no significant difference between the pretest scores for the *DQN* ($M = 59.23$, $SD = 30.63$) and the *Expert* conditions ($M = 57.42$, $SD = 30.95$): $t(82) = 0.27$, $p = 0.79$. For **S19**, 83 students were randomly assigned to three conditions through stratified sampling: *DQN + Explanation* (*DQN+Exp*) (N = 30), *Student Choice* (N = 30), and the *Expert* baseline (N = 23). In the Student Choice condition, once a next problem is presented the students will make decisions on whether they want *the ITS to show* them how to solve the next problem (WE) or they want *to solve* the next problem themselves (PS). A one-way ANOVA test showed no significant difference in the pretest scores among the three conditions: $F(1, 81) = 0.26, p = 0.61$. More specifically, we have *DQN+Exp* ($M = 54.2$, $SD = 30.0$), *Student Choice* ($M = 50.3$, $SD = 31.3$), and *Expert* Baseline ($M = 49.9$, $SD = 35.8$). In short, our results suggested that all conditions were balanced in incoming competence in both F18 and S19.

6 Results

6.1 F18 Study:

Overall, no significant difference was found on the posttest between DQN ($M = 48.6, SD = 22.7$) and Expert-Baseline ($M = 54.0, SD = 18.3$). A one-way ANCOVA analysis on posttest scores using Condition as factor and pretest scores as a covariate shows that there was no significant difference: $F(1, 81) = 1.76, p = 0.19$. Moreover, much to our surprise, no significant differences were found on the total training time nor on the total number of WE and PS students experienced between the two conditions. So, our DQN-induced bath DRL policy is as effective as the Expert baseline policy.

Table 1. Results of S19 study by condition.

	PostTest	Training Time (mins.)	PS Count	WE Count
DQN+Exp	**41.61** (25.07)	93.0 (109.6)	9.40 (2.42)	6.10 (1.21)
Student Choice	34.24 (20.09)	75.5 (104.0)	8.06 (3.15)	7.46 (2.14)
Expert Baseline	29.44 (16.43)	65.8 (87.7)	8.13 (1.74)	7.34 (1.26)

6.2 S19 Study:

The S19 study had two goals: one was to determine whether DQN with simple explanations (DQN+Exp) can be more effective than the Expert baseline policy, and the other was to determine whether Student Choice can be more effective than either the DQN+Exp or the Expert baseline policy. In the following, we will first compare the three conditions in terms of learning performance and then perform a log analysis. Table 1 shows a comparison of the posttest, total training time, the total number of PSs, and the total number of WEs among the three conditions, showing the mean (and SD) for each value.

Learning Performance A one-way ANOVA test using the condition as a factor showed a significant difference in the posttest scores: $F(1, 81) = 4.47, p = 0.037$, with means (SD) shown in the first column in Table 1 for each condition. Furthermore, a one-way ANCOVA analysis on posttest scores using Condition as factor and pretest scores as a covariate confirms a significant difference in the posttest scores: $F(1, 80) = 4.25, p = 0.042$. Contrast analysis revealed that the *DQN+Exp* condition significantly outperformed the *Expert* condition: $t(79) = 2.02, p = 0.046$; but no significant difference was found between the *DQN+Exp* and *Student Choice* conditions: $t(79) = 1.30, p = 0.20$ or between the *Student Choice* and *Expert* conditions: $t(79) = 0.81, p = 0.42$. In short, our results showed that on the posttest scores, *DQN+Exp* significantly our-performs the *Expert* condition, and no significant difference was found between the *Student Choice* and *Expert* conditions.

Training Time and Log Analysis The second column in Table 1 shows the average amount of total training time (in minutes) students spent on the tutor for each condition. Despite the differences among the three conditions, a one-way ANOVA test using the condition as a factor showed no significant difference in time on task among them: $F(1, 81) = 0.97, p = 0.33$.

The last two columns in Table 1 show the average number of WEs and PSs that each condition experienced in S19. When comparing the *DQN+Exp* and the *Expert* conditions, a t-test showed a significant difference in the number of PS: $t(51) = 2.22, p = 0.031$, and a significant difference in the number of WE: $t(51) = 3.62, p = 0.0007$, with the *DQN* condition seeing about one more PS and one less WE than the *Expert* condition. When comparing the the *DQN+Exp* and *Student Choice* conditions, a t-test showed a marginal difference in the number of PS $t(58) = 1.84, p = 0.07$, and a significant difference in the number of WE $t(58) = -3.04, p = 0.003$, with the *DQN+Exp* condition seeing about one more PS and one less WE than the Student Choice group. A contrast analysis also showed a significant difference in the number of PS ($t(80) = 2.02, p = 0.047$) and in the number of WE $t(80) = -3.26, p = 0.001$ between the *DQN+Exp* and *Student Choice* conditions.

Much to our surprise, the *Student Choice* condition behaved in a very similar way to the *Expert* condition in that no significant difference was found between the two conditions on the number of PS: $t(51) = -0.09, p = 0.93$. Similarly, no difference was found on the number of WE: $t(51) = -0.251, p = 0.802$. To summarize, our log analysis shows that *DQN+Exp* generated more PS and less WE than the other two conditions and no significant difference was found between the *Student Choice* and *Expert* conditions.

7 Discussion and Conclusion

This work demonstrates one potential way to combine data-driven methods such as DRL with other educational strategies that increase student autonomy and agency, and observe that it can benefit student learning in our Intelligent Tutoring System. In this work, we investigated the impact of 1) providing students with simple explanations for the decisions of a batch DRL policy and 2) the impact of students' pedagogical decision-making on learning. We focused on whether to give students a WE or to engage them in PS. We strictly controlled the domain content to isolate the impact of *pedagogy* from *content*.

In two classroom studies, we compared the batch DRL policy (with and without explanations), the Student Choice pedagogical decision making and the Expert baseline. Overall, our results show that when deciding whether to approach the next problem as PS or WE, both batch DRL-induced policies and Student Choice can be as effective as the Expert baseline policy; however by combining batch DRL-induced policies with simple explanations, we can significantly improve students' learning performance more than our expert-designed baseline policy. One potential hypothesis is that simple explanations can promote students' buy-in to pedagogical decisions made by batch DRL induced policies. However, further survey studies are needed to determine this hypothesis. Interestingly, our results showed that students can make as effective problem-level decisions as the Expert baseline policy. Surprisingly, students selected as many PSs and WEs as the Expert policy but the variance of decisions in Student Choice is larger than those of Expert Baseline.

We believe that the results from this research can shed some light on how to apply DRL for human-centric tasks such as an ITS, and further research is needed to fully understand why simple explanations work and whether they can indeed be applied effectively to other domains. Furthermore, in this work, we have only explored straightforward, human-expert designed explanations, which can sometimes be limiting. In the future, personalized, data-driven explanations, will make the system more powerful and provide more accurate explanations.

Acknowledgements: This research was supported by the NSF Grants: #1726550, #1651909 and #1937037.

References

1. Aleven, V., Koedinger, K.R.: Limitations of student control: Do students know when they need help? In: Gauthier, G., Frasson, C., VanLehn, K. (eds.) Intelligent Tutoring Systems. pp. 292–303. Springer Berlin Heidelberg, Berlin, Heidelberg (2000)
2. Anderson, J.R., Corbett, A.T., Koedinger, K.R., Pelletier, R.: Cognitive tutors: Lessons learned. The journal of the learning sciences **4**(2), 167–207 (1995)
3. Andrychowicz, M., Baker, B., et al.: Learning dexterous in-hand manipulation. arXiv preprint arXiv:1808.00177 (2018)
4. Azizsoltani, H., Kim, Y.J., Ausin, M.S., Barnes, T., Chi, M.: Unobserved is not equal to non-existent: using gaussian processes to infer immediate rewards across contexts. In: Proceedings of the 28th International Joint Conference on Artificial Intelligence. pp. 1974–1980. AAAI Press (2019)
5. Barria-Pineda, J., Brusilovsky, P.: Making educational recommendations transparent through a fine-grained open learner model. In: IUI Workshops (2019)
6. Beck, J., Woolf, B.P., Beal, C.R.: Advisor: A machine learning architecture for intelligent tutor construction. AAAI/IAAI **2000**(552-557), 1–2 (2000)
7. Chi, M., VanLehn, K., Litman, D., Jordan, P.: 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)
8. Chi, M., VanLehn, K., Litman, D., Jordan, P.: An evaluation of pedagogical tutorial tactics for a natural language tutoring system: A reinforcement learning approach. International Journal of Artificial Intelligence in Education **21**(1-2), 83–113 (2011)
9. Cordova, D.I., Lepper, M.R.: Intrinsic motivation and the process of learning: Beneficial effects of contextualization, personalization, and choice. Journal of educational psychology **88**(4), 715 (1996)
10. Deci, E.L., Eghrari, H., Patrick, B.C., Leone, D.R.: Facilitating internalization: The self-determination theory perspective. Journal of personality **62**(1), 119–142 (1994)
11. Doroudi, S., Holstein, K., Aleven, V., Brunskill, E.: Towards understanding how to leverage sense-making, induction and refinement, and fluency to improve robust learning. International Educational Data Mining Society (2015)
12. Eslami, M., Krishna Kumaran, S.R., Sandvig, C., Karahalios, K.: Communicating algorithmic process in online behavioral advertising. In: Proceedings of the 2018 CHI conference on human factors in computing systems. pp. 1–13 (2018)
13. Fujimoto, S., Conti, E., Ghavamzadeh, M., Pineau, J.: Benchmarking batch deep reinforcement learning algorithms. arXiv preprint arXiv:1910.01708 (2019)
14. Hasselt, H.V.: Double q-learning. In: Advances in Neural Information Processing Systems. pp. 2613–2621 (2010)
15. Iglesias, A., Martínez, P., Aler, R., Fernández, F.: Learning teaching strategies in an adaptive and intelligent educational system through reinforcement learning. Applied Intelligence **31**(1), 89–106 (2009)
16. Iglesias, A., Martínez, P., Aler, R., Fernández, F.: Reinforcement learning of pedagogical policies in adaptive and intelligent educational systems. Knowledge-Based Systems **22**(4), 266–270 (2009)
17. Jang, H.: Supporting students' motivation, engagement, and learning during an uninteresting activity. Journal of Educational Psychology **100**(4), 798 (2008)
18. Jaques, N., Ghandeharioun, A., Shen, J.H., Ferguson, C., Lapedriza, A., Jones, N., Gu, S., Picard, R.: Way off-policy batch deep reinforcement learning of implicit human preferences in dialog. arXiv preprint arXiv:1907.00456 (2019)

19. Kinzie, M.B., Sullivan, H.J.: Continuing motivation, learner control, and cai. Educational Technology Research and Development **37**(2), 5–14 (1989)
20. Koedinger, K.R., Anderson, J.R., Hadley, W.H., Mark, M.A.: Intelligent tutoring goes to school in the big city. International Journal of Artificial Intelligence in Education (IJAIED) **8**, 30–43 (1997)
21. Koedinger, K.R., Brunskill, E., Baker, R.S., McLaughlin, E.A., Stamper, J.: New potentials for data-driven intelligent tutoring system development and optimization. AI Magazine **34**(3), 27–41 (2013)
22. Kohn, A.: Choices for children. Phi Delta Kappan **75**(1), 8–20 (1993)
23. Kunkel, J., Donkers, T., Michael, L., Barbu, C.M., Ziegler, J.: Let me explain: Impact of personal and impersonal explanations on trust in recommender systems. In: Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems. pp. 1–12 (2019)
24. Lange, S., Gabel, T., Riedmiller, M.: Batch reinforcement learning. In: Reinforcement learning, pp. 45–73. Springer (2012)
25. Mandel, T., Liu, Y.E., Levine, S., Brunskill, E., Popovic, Z.: Offline policy evaluation across representations with applications to educational games. In: Proceedings of the 2014 international conference on Autonomous agents and multi-agent systems. pp. 1077–1084. International Foundation for Autonomous Agents and Multiagent Systems (2014)
26. McLaren, B.M., van Gog, T., et al.: Exploring the assistance dilemma: Comparing instructional support in examples and problems. In: Intelligent Tutoring Systems. pp. 354–361. Springer (2014)
27. McLaren, B.M., Isotani, S.: When is it best to learn with all worked examples? In: AIED. pp. 222–229. Springer (2011)
28. McLaren, B.M., Isotani, S.: When is it best to learn with all worked examples? In: International Conference on Artificial Intelligence in Education. pp. 222–229. Springer (2011)
29. McLaren, B.M., Lim, S.J., Koedinger, K.R.: When and how often should worked examples be given to students? new results and a summary of the current state of research. In: Proceedings of the 30th annual conference of the cognitive science society. pp. 2176–2181 (2008)
30. Mnih, V., Kavukcuoglu, K., Silver, D., et al.: Human-level control through deep reinforcement learning. Nature **518**(7540), 529 (2015)
31. Najar, A.S., Mitrovic, A.: Learning with intelligent tutors and worked examples: selecting learning activities adaptively leads to better learning outcomes than a fixed curriculum. UMUAI **26**(5), 459–491 (2016)
32. Palanca, J., Heras, S., Rodríguez Marín, P., Duque, N., Julián, V.: An argumentation-based conversational recommender system for recommending learning objects. In: Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems. pp. 2037–2039 (2018)
33. Precup, D., Sutton, R.S., Singh, S.P.: Eligibility traces for off-policy policy evaluation. In: ICML. pp. 759–766. Citeseer (2000)
34. Rafferty, A.N., Brunskill, E., et al.: Faster teaching via pomdp planning. Cognitive science **40**(6), 1290–1332 (2016)
35. Rago, A., Cocarascu, O., Toni, F.: Argumentation-based recommendations: Fantastic explanations and how to find them (2018)
36. Reeve, J., Jang, H., Hardre, P., Omura, M.: Providing a rationale in an autonomy-supportive way as a strategy to motivate others during an uninteresting activity. Motivation and emotion **26**(3), 183–207 (2002)

37. Renkl, A., Atkinson, R.K., et al.: From example study to problem solving: Smooth transitions help learning. The Journal of Experimental Education **70**(4), 293–315 (2002)
38. Roll, I., Baker, R.S.d., Aleven, V., Koedinger, K.R.: On the benefits of seeking (and avoiding) help in online problem-solving environments. Journal of the Learning Sciences **23**(4), 537–560 (2014)
39. Rowe, J., Mott, B., Lester, J.: Optimizing player experience in interactive narrative planning: a modular reinforcement learning approach. In: Tenth Artificial Intelligence and Interactive Digital Entertainment Conference (2014)
40. Rowe, J.P., Lester, J.C.: Improving student problem solving in narrative-centered learning environments: A modular reinforcement learning framework. In: AIED. pp. 419–428. Springer (2015)
41. Salden, R.J., Aleven, V., Schwonke, R., Renkl, A.: The expertise reversal effect and worked examples in tutored problem solving. Instructional Science **38**(3), 289–307 (2010)
42. Sansone, C., Weir, C., Harpster, L., Morgan, C.: Once a boring task always a boring task? interest as a self-regulatory mechanism. Journal of personality and social psychology **63**(3), 379 (1992)
43. Sansone, C., Wiebe, D.J., Morgan, C.: Self-regulating interest: The moderating role of hardiness and conscientiousness. Journal of personality **67**(4), 701–733 (1999)
44. Schraw, G., Flowerday, T., Reisetter, M.F.: The role of choice in reader engagement. Journal of Educational Psychology **90**(4), 705 (1998)
45. Schwab, D., Ray, S.: Offline reinforcement learning with task hierarchies. Machine Learning **106**(9-10), 1569–1598 (2017)
46. Schwonke, R., Renkl, A., Krieg, C., Wittwer, J., Aleven, V., Salden, R.: The worked-example effect: Not an artefact of lousy control conditions. Computers in Human Behavior **25**(2), 258–266 (2009)
47. Shen, S., Chi, M.: Aim low: Correlation-based feature selection for model-based reinforcement learning. International Educational Data Mining Society (2016)
48. Shen, S., Chi, M.: Reinforcement learning: the sooner the better, or the later the better? In: Proceedings of the 2016 Conference on User Modeling Adaptation and Personalization. pp. 37–44. ACM (2016)
49. Shyu, H.Y., Brown, S.W.: Learner control versus program control in interactive videodisc instruction: What are the effects in procedural learning. International Journal of Instructional Media **19**(2), 85–95 (1992)
50. Silver, D., Huang, A., Maddison, C.J., et al.: Mastering the game of go with deep neural networks and tree search. nature **529**(7587), 484 (2016)
51. Silver, D., Hubert, T., Schrittwieser, J., et al.: A general reinforcement learning algorithm that masters chess, shogi, and go through self-play. Science **362**(6419), 1140–1144 (2018)
52. Sweller, J., Cooper, G.A.: The use of worked examples as a substitute for problem solving in learning algebra. Cognition and Instruction **2**(1), 59–89 (1985)
53. Tsai, C.H., Brusilovsky, P.: Designing explanation interfaces for transparency and beyond. In: IUI Workshops (2019)
54. Van Hasselt, H., Guez, A., Silver, D.: Deep reinforcement learning with double q-learning. In: AAAI. vol. 2, p. 5. Phoenix, AZ (2016)
55. VanLehn, K., Graesser, A.C., et al.: When are tutorial dialogues more effective than reading? Cognitive science **31**(1), 3–62 (2007)
56. Vinyals, O., Babuschkin, I., Czarnecki, W., et al.: Grandmaster level in starcraft ii using multi-agent reinforcement learning. Nature **575**, 350 (2019)

57. Wang, P., Rowe, J., Min, W., Mott, B., Lester, J.: Interactive narrative personalization with deep reinforcement learning. In: Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence (2017)
58. Yeh, S.W., Lehman, J.D.: Effects of learner control and learning strategies on english as a foreign language (efl) learning from interactive hypermedia lessons. Journal of Educational Multimedia and Hypermedia **10**(2), 141–159 (2001)