Online Reinforcement Learning-Based Pedagogical Planning for Narrative-Centered Learning Environments

Fahmid Morshed Fahid[1], **Jonathan Rowe**[1], **Yeojin Kim**[1],
Shashank Srivastava[2], **James Lester**[1]

[1]North Carolina State University
[2]University of North Carolina Chapel Hill

ffahid@ncsu.edu, jprowe@ncsu.edu, ykim32@ncsu.edu, ssrivastava@cs.unc.edu, lester@ncsu.edu

Abstract

Pedagogical planners can provide adaptive support to students in narrative-centered learning environments by dynamically scaffolding student learning and tailoring problem scenarios. Reinforcement learning (RL) is frequently used for pedagogical planning in narrative-centered learning environments. However, RL-based pedagogical planning raises significant challenges due to the scarcity of data for training RL policies. Most prior work has relied on limited-size datasets and offline RL techniques for policy learning. Unfortunately, offline RL techniques do not support on-demand exploration and evaluation, which can adversely impact the quality of induced policies. To address the limitation of data scarcity and offline RL, we propose INSIGHT, an online RL framework for training data-driven pedagogical policies that optimize student learning in narrative-centered learning environments. The INSIGHT framework consists of three components: a narrative-centered learning environment simulator, a simulated student agent, and an RL-based pedagogical planner agent, which uses a reward metric that is associated with effective student learning processes. The framework enables the generation of synthetic data for on-demand exploration and evaluation of RL-based pedagogical planning. We have implemented INSIGHT with OpenAI Gym for a narrative-centered learning environment testbed with rule-based simulated student agents and a deep Q-learning-based pedagogical planner. Our results show that online deep RL algorithms can induce near-optimal pedagogical policies in the INSIGHT framework, while offline deep RL algorithms only find suboptimal policies even with large amounts of data.

Introduction

Narrative-centered learning environments utilize game engine technologies to create effective and engaging learning opportunities for students through interactive narratives with believable characters, plots, and immersive virtual worlds (Mayer 2019; Naul and Liu 2020). These environments can provide personalized pedagogical support in terms of targeted feedback, hints, and explanations to foster enhanced learning in story-based problem-solving scenarios. Learning an effective pedagogical policy can be challenging as student behaviors and needs can vary, and there is no single pedagogical support that is appropriate for all circumstances. Pedagogical planning is the task of creating pedagogical plans for enacting a range of decisions, such as sequencing problems, intervening in problem-solving, and offering explanations, feedback, and hints to support students' learning (VanLehn 2006; Woolf 2008).

Reinforcement learning (RL), a type of machine learning paradigm for sequential decision making that maximizes long-term reward (Sutton and Barto 2018), is well suited for pedagogical planning. However, RL relies on large amounts of data with rewards to learn optimal policies (Arulkumaran et al. 2017). Collecting students' interaction data and learning experience data can be expensive as classroom studies typically involve a couple hundred students at most. Most prior work on RL-based pedagogical planning has relied on limited-sized datasets and *offline RL* (aka *batch RL*) (Singla et al. 2021), a special type of RL technique that uses prior data without further exploration to find optimal policies by estimating transitions and rewards.

Although this approach has shown significant promise (Ju et al. 2022; Zhou et al. 2022), offline RL has limited capabilities in terms of learning and evaluating policies. With relatively small datasets and no exploration, offline RL can yield suboptimal policies (Prudencio et al. 2023). Moreover, the induced policies are typically evaluated using off-policy evaluation (OPE) techniques, which can produce unreliable and noisy results (Uehara, Shi, and Kallus 2022; Fu et al. 2021). An alternative approach is to train online RL-based policies (i.e., RL that leverages on-demand exploration and evaluation) using synthetic data by simulating student behaviors in the learning environment.

In this work, we propose a reinforcement learning-based framework for pedagogical planning in narrative-centered

learning environments, Intelligent Narrative System for RL-Based Interactive Guidance and Helpful Tutoring (INSIGHT). INSIGHT provides opportunities for devising pedagogical policies for narrative-centered learning environments using online RL techniques. INSIGHT consists of three components: (1) a simulator of a narrative-centered learning environment that is deterministic and produces immediate rewards, (2) a simulated student agent that can interact with the simulator to generate synthetic data, and (3) an RL-based pedagogical planner agent that dynamically induces pedagogical policies by altering the problem-solving scenario and scaffolding of the learning environment simulator. The RL process is guided by a reward signal that is designed to reflect the quality of students' learning experiences and, importantly, can be calculated directly based on observing a simulated student's interactions with the simulated environment.

We implemented INSIGHT as an OpenAI Gym (Brockman et al. 2016) environment that emulates a testbed narrative-centered learning environment for middle school microbiology education, CRYSTAL ISLAND (Rowe et al. 2011). We used a previously validated in-game scores as the basis for calculating RL rewards and a rule-based simulated student agent to generate synthetic data for training RL-based pedagogical planners. Using INSIGHT, we trained an online deep RL-based pedagogical planner with double deep Q-networks (DQN) (Van Hasselt, Guez, and Silver 2016) and compared it against multiple offline RL-based planners with varying sizes of training datasets. Results showed that the INSIGHT framework supports the use of online RL to find near-optimal pedagogical policies, whereas offline RL approaches, such as offline double DQN, batch-conservative Q-learning (BCQ) (Fujimoto, Merger, and Precup 2019), and constrained Q-learning (CQL) (Kumar et al. 2020), induce suboptimal policies even with reasonably large training datasets.

Related Work

A wide range of computational approaches has been investigated for devising pedagogical policies in different learning environments. The most common approach is rule-driven pedagogical planning where pedagogical support is provided based on expert-designed production rules such as event-driven rules, time-driven rules, score-based rules, and constraint-based rules (Azevedo et al. 2022; Johnson et al. 2019; Lindberg and Laine 2018; Mitrovic and Ohlsson 2016). Others investigated probabilistic reasoning (Hooshyar et al. 2021; Long and Aleven 2017; Shute et al. 2021) and supervised learning (Munshi et al. 2022; Wiggins et al. 2015) for providing pedagogical support to students. However, these approaches require extensive expert knowledge, or they provide support based on immediate need without contextualizing long-term learning goals.

Recent years have seen the use of RL as an alternative approach to devising effective pedagogical policies (Doroudi, Aleven, and Brunskill 2019; Singla et al. 2021). Much of the prior work on RL-based pedagogical planning has relied on offline RL approaches. For example, Cai et al. (2021) used offline multi-arm bandits with crowd-sourced data to learn a math-based pedagogical planner. Several studies found that immediate rewards can help with policy learning in terms of convergence and rewards in offline RL (Ausin et al. 2021; Azizsoltani et al. 2019; Fahid et al. 2022). Others investigated students' agency in pedagogical decisions using offline RL (Ju et al. 2022). Some studies have investigated offline RL-based pedagogical planning in narrative-centered learning environments. Rowe et al. (2014) and Wang et al. (2016) investigated offline RL with modular architectures to induce pedagogical policies. Sawyer et al. (2017) used offline RL with multi-objective Markov decision processes (MDPs) to induce pedagogical policies that optimize students' learning and engagement.

In online RL-based pedagogical planning, early research relied on classical RL techniques such as policy iteration (Martin and Arroyo 2004) and value iterations (Iglesias et al. 2008) with simulated students. Other work has investigated deep RL techniques with online RL techniques. Mu et al. (2022) used deep knowledge tracing to generate synthetic data for online deep RL-based pedagogical planning. Similar approaches have been taken by others (Bassen et al. 2020; Zhang et al. 2022). Rafferty et al. (2016) used hand-designed student behaviors to devise an online RL-based pedagogical planner. Ruan et al. (2023) combined online and offline RL to induce pedagogical planners for a math learning environment. Wang et al. (2018) introduced bipartite LSTMs for inducing online deep RL-based pedagogical planners for narrative-centered learning environments by creating synthetic student interaction data and learning outcome data. Most of these approaches simulate student behaviors by directly estimating or hand-designing transitions and rewards with minimal validation.

There has been limited work investigating online RL-based pedagogical planners for narrative-centered learning environments that emphasize the validity of the student interactions in the synthetic training data or the validity of the rewards used to guide the RL process. Our work presents a framework for online RL-based pedagogical planning in narrative-centered learning environments.

INSIGHT Framework

To learn effective RL-based pedagogical planners, we created INSIGHT, an online RL framework for narrative-centered learning environments. INSIGHT includes three

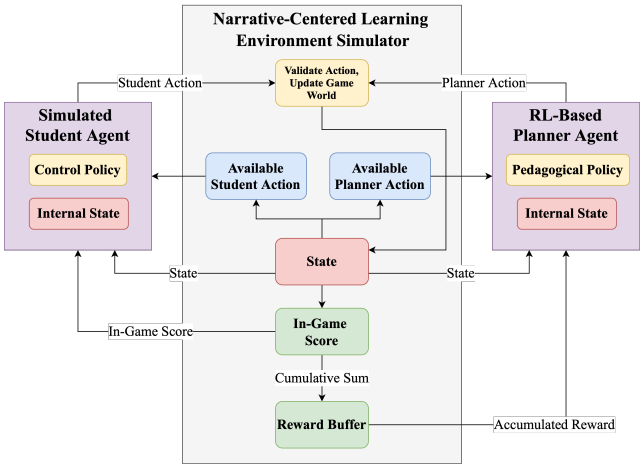

Figure 1: INSIGHT Online RL-Based Pedagogical Planning Framework

components: 1) a narrative-centered learning environment simulator (henceforth referred to as the *simulator*), 2) a simulated student agent (henceforth referred to as the *student agent*), and 3) an RL-based pedagogical planner agent (henceforth referred to as the *planner agent*).

The student agent and the planner agent are designed as modular components with their own internal states and decision policies (see Figure 1). The student agent can enact *student actions* ($A_{student}$) in the simulator based on its own internal policy and current state to emulate a wide range of student learning behaviors. Similarly, the planner agent can perform a given set of *planner actions* ($A_{planner}$) based on its own internal policy and current state. In the simulator, the set of available actions for the student agent ($A_{student}$) and the planner agent ($A_{planner}$) are determined by the rules of the narrative-centered learning environment.

The simulator incorporates in-game scores as rewards that reflect students' learning in the learning environment. Typically, in-game score metrics are expert-designed, and prior work has shown that in-game scores can correlate with key learning processes and outcomes (Nietfeld, Shores, and Hoffmann 2014; Cheng, Lin, and She 2015; Westera et al. 2019). In-game score serves as the basis for calculating immediate rewards for RL-based pedagogical planning in INSIGHT. Different student actions ($A_{student}$) generate in-game scores that are kept as a cumulative sum in a buffer as immediate rewards (see Figure 1). When control is given to the planner agent, these immediate rewards are provided to the planner to guide the RL process for inducing an effective pedagogical policy. As the task of the planner agent is to support students, no reward is gathered by the planner directly. For example, the planner may provide a hint suggesting that a student should perform a particular action, but the student is responsible for carrying out the action. These actions produce corresponding rewards (in-game score points), which are then provided to the planner to guide the policy learning.

The simulator controls the game logic, along with the state of the game world, the transition rules, and the rewards in a manner that mirrors that of a narrative-centered learning environment; analogous to a digital twin (VanDerHorn and Mahadevan 2021). Moreover, the simulator introduces a level of abstraction over the gameplay mechanics, student actions, and planner actions while maintaining the original environment's core gameplay design. For example, the simulator hides the actual time passed in the game by using an event-driven timeline. At each step, the simulator determines which agent (i.e., the student agent or the planner agent) control flow should be given to, the set of possible actions, and the current state of the game world. Either the student agent or the planner agent takes as input the current state of the game world, the set of possible actions, and any incremental rewards that have been accrued. The agent then selects an action according to its own internal policy and state. After an action is taken by an agent, the simulator validates the action, updates the game world, and produces a reward when available. Note that the state of the simulator and the rewards are deterministically calculated based on the student's actions or the planner's actions, and the simulator is responsible for ensuring that each transition and reward is valid in the learning environment. However, from the planner's perspective, the behavior of the system is non-deterministic and noisy. For example, two students in identical states can take different student actions and end up in different states (with different rewards) even though they received the same decisions from the pedagogical planner.

Prototype Implementation of INSIGHT

This section describes the implementation of the INSIGHT framework with a narrative-centered learning environment testbed for middle school microbiology education. First, we describe the implementation of INSIGHT, including the in-game score metric and adaptable event sequences that are utilized to instantiate the pedagogical planning task. Next, we describe our student agent. Finally, we discuss the online RL-based pedagogical planning architecture. The complete implementation is made using the OpenAI Gym environment for rapid prototyping and experimentation.

Testbed Learning Environment

We implemented INSIGHT with a testbed narrative-centric learning environment based on the CRYSTAL ISLAND learning environment for middle school microbiology (Rowe et al. 2011). The testbed features a science mystery about an infectious outbreak on a remote island. Students play the role of a medical detective investigating the outbreak and are responsible for identifying the disease and

Figure 2: Testbed narrative-centered learning environment based on CRYSTAL ISLAND.

its source of transmission in the virtual world (see Figure 2). The game randomizes the disease (between salmonella and influenza) and the transmission source (foodborne or airborne) at the start of the game. Students can move between different locations on the island, talk to different non-player characters (NPCs), read books and posters, collect different objects as samples, test them for possible contaminants using different testing kits (e.g., influenza test, food pathogen test), complete a diagnosis worksheet, and submit their diagnosis to attempt to solve the science mystery. These are the set of available student actions ($A_{student}$). There are total four locations (lab, infirmary, dining hall, and outdoors), four non-player characters (Kim, Teresa, Robert, Elise), 14 books and posters, 10 different objects, 7 diseases, and 5 different test kits. The problem-solving scenario ends when the student finds the correct diagnosis or more than 200 timesteps have elapsed.

In-Game Score

CRYSTAL ISLAND's in-game score is incremented after key in-game milestones are completed by students, and it is decremented when students make errors or enact unproductive learning behaviors during gameplay. The in-game score metric was carefully designed and empirically validated in prior studies with middle school students; it was found to be correlated with student motivation and learning processes in CRYSTAL ISLAND (Rowe et al. 2010; Nietfeld, Shores, and Hoffmann 2014). The in-game score metric is designed to encourage effective learning strategies and penalize gaming-the-system. The reward model utilized in the prototype implementation of INSIGHT is lightly adapted from the in-game score metric in CRYSTAL ISLAND. Table 1 shows the in-game score metric that is used in INSIGHT's learning environment simulator.

Milestone	Scores	Recurring
Submit correct diagnosis	500	No
Solution efficiency	7500/total timesteps	No
Incorrect diagnosis	-100	Yes
Correct test performed	200	No
Incorrect test performed	-25	Yes
Talk to Kim	15/elapsed timesteps	No
Talk to Teresa	40/elapsed timesteps	No
Talk to Elise	50/elapsed timesteps	No
Talk to Robert	50/elapsed timesteps	No

Table 1: Calculation of in-game score in the testbed narrative-centered learning environment.

Adaptable Event Sequences

At predefined points during gameplay, the learning environment can alter or not alter different components of the problem scenario or scaffolding in the learning environment. We call these decision points adaptive event sequences, or AESs (Rowe et al. 2014). The simulator used

AES	Student action triggers	Effect when AES is *altered* by pedagogical planner
Story-Plot Adaptation	Test an object in the lab	The disease mutates to a seasonal variant of influenza. A newspaper is added to a virtual bulletin board that hints at the mutation. A specialized test kit is required to correctly identify the disease in this modified version of the narrative.
Character-Delivered Scaffolding	Talk to Teresa (patient) in the infirmary	Teresa reveals important information about the relevant test kit and the possible source of disease transmission.
Tool-based scaffolding	Open in-game diagnosis worksheet	Provide additional direction about how to investigate the outbreak. Specifically, the student is prompted to speak to Teresa (patient) about her medical history and about collecting a nasal swab.
Character-Behavior Directives	Move to a location after *story-plot adaptation* was enacted	Provide a virtual text message to the student that reinforces the potential for a seasonal mutation of the spreading virus.

Table 2: Adaptable event sequences (AESs) in the prototype implementation of INSIGHT

in this work features several pre-defined AESs, which serve as opportunities for dynamically altering students' story-based learning experiences during science problem solving. At each AES, there are two possible actions a pedagogical planner can take: 1) alter or 2) do not alter. Notably, AESs are designed to not interfere with the overall progression of the narrative. All possible choices by the pedagogical planner yield coherent interactive narratives that play out within the Crystal Island learning environment.

Table 2 lists the four AESs in our prototype implementation. All AESs are triggered by a corresponding student action. All AESs can occur multiple times except for the Story Plot AES, which can recur until a decision to alter the plot (i.e., introduce a mutation to the disease) is enacted. Figure 2 shows an example where the pedagogical planner enacts a decision (altered Character-Delivered Scaffolding AES) by including a new dialogue option.

Rule-Based Simulated Student Agent

We devised a simple rule-based simulated student agent that follows a random walk-based student interaction policy (3 rules for random interactions) combined with hand-designed reaction rules (4 rules for specific reactions to AES alterations with multiple sub-rules within them). These rules are governed by 6 features in students' internal states. The student's internal state consists of four features based on student actions (number of times the student asked about disease symptoms or medical history, the minimum number of times the student performed all actions, and current location of the transmission source) as well as two features denoting student individual differences: student skill (high or low) and motivation (high or low). Student skill and motivation are randomly assigned at the start of an episode and are kept constant during the episode. We designed the simulated student agent's behavior to be reflective of common student learning behaviors observed in prior studies with CRYSTAL ISLAND (Rowe et al. 2010; Rowe et al. 2014). Designing data-driven simulated student agents using machine learning is a direction for future work.

RL-Based Pedagogical Planner

To learn an optimal policy in a complex environment, a common approach is to estimate the state-action value function $Q(s, a)$ using a deep neural network (θ). The Q-function represents the expected discounted reward $G_T = \sum_t^T \gamma^t r_t$ starting from state $s_t \in S$, taking action $a_t \in A_{planner}$, and following a policy π where γ is the discount factor. This is known as a deep Q-network or DQN (Mnih et al. 2015). For our online RL-based pedagogical planner, we use double DQNs (Van Hasselt, Guez, and Silver 2016), a variant of DQNs that reduces the overestimation bias by separating the action selection and action evaluation components of the model into two separate networks, namely, a target network $(\bar{\theta})$ and an online network (θ). The following Bellman equation is used to induce an optimal $Q^*(s_t, a_t)$ function by iteratively sampling states, actions, and rewards from a finite experience buffer:

$$Q(s_t, a_t) = r_t + \gamma Q\left(s_{t+1}, argmax_{a_{t+1} \in A_{planner}} Q(s_{t+1}, a_{t+1}; \theta); \bar{\theta}\right) \quad (1)$$

States, Actions, and Rewards

RL problems are typically defined using an MDP formalism with a state-action-reward paradigm. This section describes the MDP utilized for this work.

State S: The state representation consists of 7 binary features and 6 numerical features. The binary features are student skill (high/low), transmission source location (backpack or not), knowing about mutation, and the type of AES that is triggered (four binary indicators). The numerical features are the number of times the student submitted a proposed solution, talked about symptoms, talked about medical history, learned about a possible disease mutation, the number of times each AES was altered, and the minimum number of times any one type of student action was performed across the entire student learning episode.

Action $A_{planner}$: Each time an AES is triggered, there are two distinct planner actions: alter and do not alter. These alternatives manifest as follows: 1) Story Plot Adaptation AES: the decision involves mutating the disease or not; 2) Character-Delivered Scaffolding AES: the decision pertains to providing or not providing a hint through one of the NPC; 3) Tool-Based Scaffolding AES: the decision pertains to providing or not providing a hint through the in-game diagnosis worksheet; and 4) Character-Behavior Directive AES: the decision involves prompting or not prompting a character to send a virtual text message to the student about disease mutation. Although the alter/no-alter decision has different meanings based on the different AESs, we model them as unified actions under a single RL framework by incorporating the AES-trigger within the state features.

Reward R: Immediate rewards are calculated by taking the cumulative change in the student's in-game score between the current planner action and the next planner action (or the end of the episode). This value is assigned as a reward to the current planner action. The rewards are calculated by taking the cumulative sum of in-game score points based on sequences of student actions. We ignore any rewards collected prior to the first planner action, as those rewards are irrelevant for pedagogical policy learning.

Evaluation

For all experiments, we keep the rule-based simulated student agent and the narrative-centered learning environment simulator fixed to generate synthetic data and

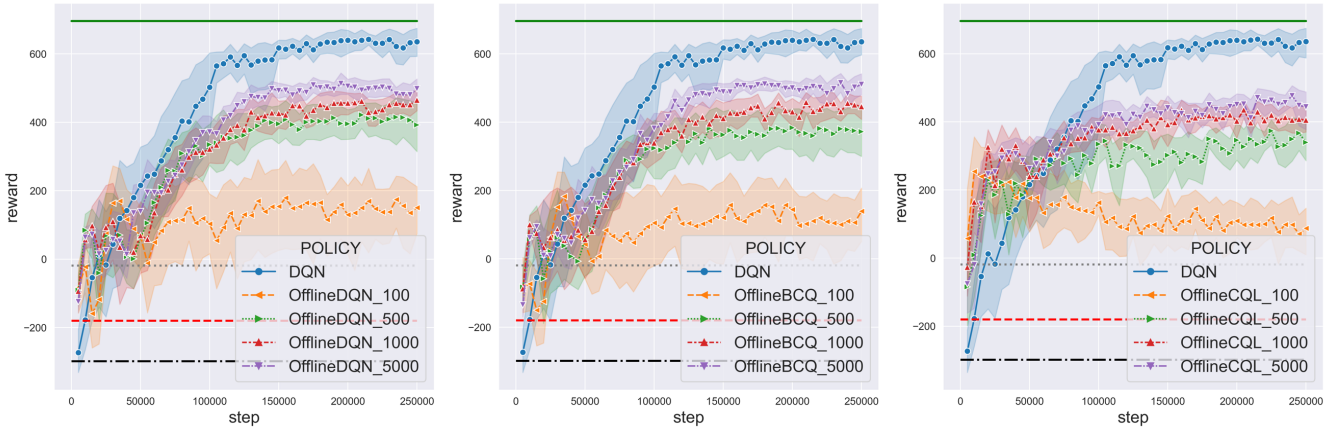

Figure 3: Comparing results of online double DQNs (DQN) against offline DQNs (left), offline BCQ (middle), and offline CQL (right) with varying sizes of training data. The green, red, gray, and black horizontal lines represent the *optimal*, *random*, *always-alter*, and *always-do-not-alter* policies, respectively.

evaluate the induced pedagogical planning policies. The only component of the framework that is manipulated is the algorithm used to induce pedagogical planner policies. Also, to simulate different types of students, at the start of each episode we randomly selected a student skill level (from *high/low*) and student motivation level (from *high/low*). We evaluate pedagogical planning policies' performance every 5,000 steps by simulating 100 episodes with 100 random rule-based student agents with the current policy. We report the mean of total rewards for each student.

To evaluate the INSIGHT framework, we use double DQNs, as it provides stability in learning and converges more efficiently than vanilla DQNs. We utilize two hidden layers with 256 neurons, a learning rate of 0.0005, a discount factor of 0.99, a buffer size of 10,000, and a batch size of 256. We update the target network after 1,000 steps. We use a 3-step temporal difference for Q-learning for better stability. We refer to this architecture simply as "DQN."

To compare the INSIGHT framework for inducing online RL-based pedagogical planners against offline RL techniques, we use offline versions of the same DQN architecture (offline DQN), along with constrained Q-learning (offline CQL) and batch-constrained Q-learning (offline BCQ). We keep the network architectures and hyperparameters of the offline RL similar to the online DQNs. Additionally, for BCQ, we use the action-constrained probability threshold $\tau = 0.3$ and for CQL, we use constrained constant $\alpha = 1$. We explore training with 100 episodes, 500 episodes, 1,000 episodes, and 5,000 episodes of prior data (i.e., synthetic data sampled using a uniform random policy and rule-based student in INSIGHT) for each of the offline RL algorithms. For all experiments, we use the d3rlpy library (Seno and Imai 2022).

We train all RL-based planners for 250,000 steps and repeat all experiments 10 times with different random seeds. To support the evaluation of RL-based policies, we compare them to four baseline policies: 1) a hand-designed "optimal" policy which has a mean reward of 696 (green line), 2) a random policy which chooses actions from a uniform random distribution with a mean reward of -180 (red dashed line), 3) an "always alter" policy with a mean reward of -19 (gray dotted line), and 4) an "always do not alter" policy with a mean reward of -298 (black dashed line).

Results show that online DQN performs almost as well as the optimal policy with a mean reward of 636 (*SD*=72) and converges after approximately 150,000 steps (Figure 3). Comparing online DQNs to the offline versions of DQNs (Figure 3, left), we see that with 100 training episodes (orange), offline DQNs converge relatively quickly (around 50,000 steps) but have the lowest mean reward of 150 (*SD*=109). Increasing the number of training episodes to 500 increases the overall reward to 392 (*SD*=116) and results in model convergence around 150,000 steps. Further increasing the dataset size improves performance, but the benefit is relatively small. For example, with 5,000 episodes, policies converge to a mean reward of 498 (*SD*=53). Note that this mean reward value is less than the mean reward obtained by online DQNs (636).

Similarly, we compare online DQNs to offline BCQ (Figure 3, middle). As we can see, with 100 episodes, offline BCQ converges to a mean reward of 139 (*SD*=142), similar to the offline DQN model. With increasing dataset size, performance continues to improve. With 500 episodes, BCQ converges to a mean reward of 373 (*SD*=127). With a significantly larger dataset (i.e., 5,000 episodes), BCQ converges to a mean reward of 510 (*SD*=51). Once again, this is lower than the online DQN approach.

Finally, when comparing online DQNs to offline CQL (Figure 3, right), we see similar trends. With 100 episodes (orange), CQL converges very early (around 10,000 steps) with a mean reward of 87 (*SD*=118). With increasing dataset size, improvements in performance are significant. For

example, 1,000 and 5,000 training episodes yield a mean reward of 405 (*SD*=62) and 444 (*SD*=96), respectively. Again, the performance of offline CQL is significantly lower than the online RL-based pedagogical planners.

Discussion

Results indicate that the INSIGHT framework shows substantial promise for learning and evaluating pedagogical planning policies with online RL techniques. We employed the INSIGHT framework to compare online RL-based pedagogical planners with offline RL-based planners, including some specifically crafted to address extrapolation errors associated with offline learning. Our findings indicate that offline RL algorithms can produce improved policies for pedagogical planning as more data becomes available. However, these improvements are not substantial when compared to the benefits of utilizing an online RL approach. The underlying reason for this is straightforward. Online RL-based planners have the advantage of utilizing a simulator to actively explore and exploit the environment to discover optimal policies. In contrast, offline RL-based planners rely on previously collected data, which may or may not contain sufficient samples of advantageous states and high-reward trajectories. It is interesting to note that offline DQNs demonstrate similar performance to CQL and BCQ when a sufficient amount of offline data is available (e.g., 5,000 episodes). This similarity could be attributed to the larger size dataset better representing the possible state-action pairs encountered, reducing the chance for distribution mismatch and extrapolation errors (Prudencio et al. 2023). Moreover, we demonstrated that the INSIGHT framework allows direct evaluation of different policies with cumulative rewards without relying on limited data for estimating outcomes using OPE (Levine et al. 2020).

The work presented in this study has several limitations. First, the use of a simplistic rule-based simulated student agent to learn and test pedagogical planner policies may not accurately reflect real human behaviors. However, the modular nature of the framework allows for improvements to the simulated student agent in future research. Second, the creation of the simulator involves abstracting certain low-level game mechanics. It is possible that this type of abstraction may have unexpected implications. Third, although the reward metrics were previously found to correlate with students' learning experiences (Rowe et al. 2010), it remains to be seen whether they robustly predict student learning in classroom implementations of CRYSTAL ISLAND integrated with a run-time RL-based planner. Moreover, rewards that correlate with students' long-term learning do not guarantee that they correspond to good policies (Azizsoltani et al., 2019); AESs must be carefully designed and policies must be verified to ensure they are safe and appropriate for students. Lastly, our evaluation for online RL-based pedagogical planning is limited to an investigation of double DQNs against three offline algorithms. It is possible that other offline algorithms, such as robust fitted Q-Iteration (Panaganti et al. 2022) could outperform double DQNs. Future research should explore alternative online RL approaches for pedagogical planning in narrative-centered learning environments.

Conclusion

RL-based pedagogical planning shows significant promise for enhancing student learning and engagement in narrative-centered learning environments. In this paper, we introduced INSIGHT, a multi-component online RL framework for inducing data-driven pedagogical planners in narrative-centered learning environments. INSIGHT leverages a learning environment simulator, a simulated student agent, and a pedagogical planner agent to emulate the process of a student receiving guidance from a pedagogical planner as they solve an interactive story-based problem scenario. By combining the framework with rewards that reflect effective student learning processes, we train an online RL-based pedagogical policy for a testbed narrative-centered learning environment.

Results from a series of experiments show the effectiveness of the INSIGHT framework for inducing and evaluating online RL-based pedagogical planners for narrative-centered learning environments. We compared online RL-based planners against several offline RL-based planners. Findings revealed that online RL-based planners performed nearly as well as hand-designed optimal policies, while offline RL approaches achieved suboptimal results even with significantly large training datasets. This disparity can be attributed to online RL's ability to explore and exploit state-action pairs, which are often missed by offline RLs.

This work provides several contributions, including the development of a modular framework for inducing and evaluating online RL-based pedagogical planners, the creation of a testbed RL-based pedagogical planning framework in OpenAI Gym, and an evaluation demonstrating the effectiveness of online RL-based planners. Directions for future research include enhancing the framework with more sophisticated reward models, exploring different designs of the student and planner agents, and investigating the application of INSIGHT in other narrative-centered learning environments.

Acknowledgements

This work is supported by the National Science Foundation under award DRL-2112635. Any opinions, findings, and conclusions or recommendations expressed in this material

are those of the authors and do not necessarily reflect the views of the National Science Foundation.

References

Arulkumaran, K.; Deisenroth, M.P.; Brundage, M.; and Bharath, A.A. 2017. Deep Reinforcement Learning: A Brief Survey. *IEEE Signal Processing Magazine*, 34(6): 26-38. https://doi.org/10.1109/MSP.2017.2743240

Ausin, M.S.; Azizsoltani, H.; Ju, S.; Kim, Y.J.; and Chi, M. 2021. InferNet for Delayed Reinforcement Tasks: Addressing the Temporal Credit Assignment Problem. In *Proceedings of the IEEE International Conference on Big Data*, Orlando, FL, USA, 1337-1348. https://doi.org/10.1109/BigData52589.2021.9671827

Ausin, M.S.; Maniktala, M.; Barnes, T.; and Chi, M. 2022. The Impact of Batch Deep Reinforcement Learning on Student Performance: A Simple Act of Explanation Can Go a Long Way. *International Journal of Artificial Intelligence in Education*, 1-26. https://doi.org/10.1007/s40593-022-00312-3

Azevedo, R.; Bouchet, F.; Duffy, M.; Harley, J.; Taub, M.; Trevors, G.; Cloude, E.; Dever, D.; Wiedbusch, M.; Wortha, F.; and Cerezo, R. 2022. Lessons Learned and Future Directions of MetaTutor: Leveraging Multichannel Data to Scaffold Self-Regulated Learning with an Intelligent Tutoring System. *Frontiers in Psychology*, 13: 813632.

Azizsoltani, H.; Kim, Y.; Ausin, M. S.; Barnes, T.; and Chi, M. 2019. 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*, 1974-1980.

Bassen, J.; Balaji, B.; Schaarschmidt, M.; Thille, C.; Painter, J.; Zimmaro, D.; Games, A.; Fast, E.; and Mitchell, J.C. 2020. Reinforcement Learning for the Adaptive Scheduling of Educational Activities. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, 1-12. https://doi.org/10.1145/3313831.3376518

Brockman, G.; Cheung, V.; Pettersson, L.; Schneider, J.; Schulman, J.; Tang, J.; and Zaremba, W. 2016. OpenAI Gym. *arXiv preprint arXiv:1606.01540*.

Cai, W.; Grossman, J.; Lin, Z.J.; Sheng, H.; Wei, J.T.Z.; Williams, J.J.; and Goel, S. 2021. Bandit Algorithms to Personalize Educational Chatbots. *Machine Learning*, 110(9): 2389-2418. https://doi.org/10.1007/s10994-021-05983-y

Cheng, M.T.; Lin, Y.W.; She, H.C.; and Kuo, P.C. 2017. Is Immersion of Any Value? Whether, and to What Extent, Game Immersion Experience During Serious Gaming Affects Science Learning. *British Journal of Educational Technology*, 48(2): 246-263. https://doi.org/10.1111/bjet.12386

Doroudi, S.; Aleven, V.; and Brunskill, E. 2019. Where's The Reward? A Review of Reinforcement Learning for Instructional Sequencing. *International Journal of Artificial Intelligence in Education*, 29: 568-620. https://doi.org/10.1007/s40593-019-00187-x

Fahid, F. M.; Rowe, J.; Spain, R.; Goldberg, B.; Pokorny, R.; and Lester, J. 2022. Robust Adaptive Scaffolding with Inverse Reinforcement Learning-Based Reward Design. In *Proceedings of the 23rd International Conference on Artificial Intelligence in Education*, 204–207. Springer International Publishing.

Fahid, F.M.; Rowe, J.P.; Spain, R.D.; Goldberg, B.S.; Pokorny, R.; and Lester, J. 2021. Adaptively Scaffolding Cognitive Engagement with Batch Constrained Deep Q-Networks. In *Proceedings of the 22nd International Conference on Artificial Intelligence in Education,* Utrecht, The Netherlands, 113-124. Springer International Publishing. https://doi.org/10.1007/978-3-030-78292-4_10

Fu, J.; Norouzi, M.; Nachum, O.; Tucker, G.; Wang, Z.; Novikov, A.; Yang, M.; Zhang, M.R.; Chen, Y.; Kumar, A.; and Paduraru, C. 2021. Benchmarks for Deep Off-Policy Evaluation. *arXiv preprint arXiv:2103.16596*.

Fujimoto, S.; Meger, D.; and Precup, D. 2019. Off-Policy Deep Reinforcement Learning without Exploration. In *Proceedings of the 36th International Conference on Machine Learning*, Long Beach, CA, USA, 2052-2062. PMLR.

Hooshyar, D.; Malva, L.; Yang, Y.; Pedaste, M.; Wang, M.; and Lim, H. 2021. An Adaptive Educational Computer Game: Effects on Students' Knowledge and Learning Attitude in Computational Thinking. *Computers in Human Behavior*, 114: 106575. https://doi.org/10.1016/j.chb.2020.106575

Iglesias, A.; Martínez, P.; Aler, R.; and Fernández, F. 2009. Reinforcement Learning of Pedagogical Policies in Adaptive and Intelligent Educational Systems. *Knowledge-Based Systems*, 22(4), 266–270.

Johnson, E.; Lucas, G.; Kim, P.; and Gratch, J. 2019. Intelligent Tutoring System for Negotiation Skills Training. In *Proceedings of the 20th International Conference on Artificial Intelligence in Education*, Chicago, IL, USA, 122-127. Springer International Publishing. https://doi.org/10.1007/978-3-030-23207-8_23

Ju, S.; Yang, X.; Barnes, T.; and Chi, M. 2022. Student-Tutor Mixed-Initiative Decision-Making Supported by Deep Reinforcement Learning. In *Proceedings of the 23rd International Conference on Artificial Intelligence in Education*, Durham, UK, 440-452. Springer International Publishing. https://doi.org/10.1007/978-3-031-11644-5_36

Kumar, A.; Zhou, A.; Tucker, G.; and Levine, S. 2020. Conservative Q-Learning for Offline Reinforcement Learning. *Advances in Neural Information Processing Systems*, 33: 1179-1191.

Levine, S.; Kumar, A.; Tucker, G.; and Fu, J. 2020. Offline Reinforcement Learning: Tutorial, Review, and Perspectives on Open Problems. *arXiv preprint arXiv:2005.01643*.

Lindberg, R.S.; and Laine, T.H. 2018. Formative Evaluation of an Adaptive Game for Engaging Learners of Programming Concepts in K-12. *International Journal of Serious Games*, 5(2): 3-24. https://doi.org/10.17083/ijsg.v5i2.220

Long, Y.; and Aleven, V. 2017. Enhancing Learning Outcomes Through Self-Regulated Learning Support with an Open Learner Model. *User Modeling and User-Adapted Interaction*, 27(1): 55–88. https://doi.org/10.1007/s11257-016-9186-6

Martin, K. N. and Arroyo, I. 2004. AgentX: Using Reinforcement Learning to Improve the Effectiveness of Intelligent Tutoring Systems. In *Proceedings of the International Conference on Intelligent Tutoring Systems*, Berlin, 564-572.

Mayer, R.E. 2019. Computer Games in Education. *Annual Review of Psychology*, 70: 531-549. doi.org/10.1146/annurev-psych-010418-102744

Mitrovic, A; and Ohlsson, S. 2016. Implementing CBM: SQL-Tutor After Fifteen Years. *International Journal of Artificial Intelligence in Education*, 26(1):150–59. https://doi.org/10.1007/s40593-015-0049-9

Mu, T.; Theocharous, G.; Arbour, D.; and Brunskill, E. 2022. Constraint Sampling Reinforcement Learning: Incorporating Expertise for Faster Learning. In *Proceedings of the 36th AAAI Conference on Artificial Intelligence*, 36(7): 7841-7849. https://doi.org/10.1609/aaai.v36i7.20753

Munshi, A.; Biswas, G.; Davalos, E.; Logan, O.; Narasimham, G.; and Rushdy, M. 2022. Adaptive Scaffolding to Support Strategic Learning in an Open-Ended Learning Environment. *In Proceedings of the 30th International Conference on Computers in Education*, 150-156.

Naul, E.; and Liu, M. 2020. Why Story Matters: A Review of Narrative in Serious Games. *Journal of Educational Computing Research*, 58(3): 687-707. https://doi.org/10.1177/0735633119859904

Nietfeld, J. L.; Shores, L. R.; and Hoffmann, K. F. 2014. Self-regulation and Gender Within a Game-based Learning Environment. *Journal of Educational Psychology*, 106(4), 961.

Panaganti, K.; Xu, Z.; Kalathil, D.; and Ghavamzadeh, M. 2022. Robust Reinforcement Learning using Offline Data. *Advances in Neural Information Processing Systems*, 35, 32211-32224.

Prudencio, R.F.; Maximo, M.R.; and Colombini, E.L. 2023. A Survey on Offline Reinforcement Learning: Taxonomy, Review, and Open Problems. *IEEE Transactions on Neural Networks and Learning Systems*. https://doi.org/10.1109/TNNLS.2023.3250269

Rafferty, A.N.; Brunskill, E.; Griffiths, T.L.; and Shafto, P. 2016. Faster Teaching via POMDP Planning. *Cognitive Science*, 40(6): 1290-1332. https://doi.org/ 10.1111/COGS.12290

Rowe, J.; Mott, B.; and Lester, J. 2014. Optimizing Player Experience in Interactive Narrative Planning: A Modular Reinforcement Learning Approach. In *Proceedings of the 10th AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, Raleigh, NC, USA, 10(1): 160-166. https://doi.org/10.1609/aiide.v10i1.12733

Rowe, J., Shores, L., Mott, B., & Lester, J. (2011). Integrating Learning, Problem Solving, and Engagement in Narrative-centered Learning Environments. International Journal of Artificial Intelligence in Education, 21(1–2), 115–133.

Rowe, J.; Shores, L. R.; Mott, B.; and Lester, J. 2010. Individual Differences in Gameplay and Learning: A Narrative-Centered Learning Perspective. In *Proceedings of the Fifth International Conference on the Foundations of Digital Games*, 171–178. https://doi.org/10.1145/1822348.1822371

Ruan, S.; Nie, A.; Steenbergen, W.; He, J.; Zhang, J.Q.; Guo, M.; Liu, Y.; Nguyen, K.D.; Wang, C.Y.; Ying, R.; and Landay, J.A. 2023. Reinforcement Learning Tutor Better Supported Lower Performers in A Math Task. *arXiv preprint arXiv:2304.04933*.

Sawyer, R.; Rowe, J.; and Lester, J. 2017. Balancing Learning and Engagement in Game-Based Learning Environments with Multi-Objective Reinforcement Learning. In *Proceedings of 18th International Conference on Artificial Intelligence in Education*, Wuhan, China, 323-334. Springer International Publishing. https://doi.org/10.1007/978-3-319-61425-0_27

Seno, T.; and Imai, M. 2022. d3rlpy: An Offline Deep Reinforcement Learning Library. *The Journal of Machine Learning Research*, 23(1): 14205-14224.

Singla, A.; Rafferty, A.N.; Radanovic, G.; and Heffernan, N.T. 2021. Reinforcement learning for education: Opportunities and challenges. *arXiv preprint arXiv:2107.08828*.

Shute, V.J.; Smith, G.; Kuba, R.; Dai, C.P.; Rahimi, S.; Liu, Z.; and Almond, R. 2021. The Design, Development, and Testing of Learning Supports for the Physics Playground Game. *International Journal of Artificial Intelligence in Education*, 31: 357-379. https://doi.org/10.1007/s40593-020-00196-1.

Sutton, R.S.; and Barto, A.G. 2018. *Reinforcement Learning: An Introduction*. MIT Press.

Uehara, M.; Shi, C.; and Kallus, N. 2022. A Review of Off-Policy Evaluation in Reinforcement Learning. *arXiv preprint arXiv:2212.06355*.

Van Hasselt, H.; Guez, A.; and Silver, D. 2016. Deep Reinforcement Learning with Double Q-Learning. In Proceedings of the 13th AAAI Conference on Artificial Intelligence, Phoenix, Arizona, USA, 30(1): 2094-2100. https://doi.org/10.1609/aaai.v30i1.10295

VanLehn, K. 2006. The Behavior of Tutoring Systems. *International Journal of Artificial Intelligence in Education*, 16(3): 227-265.

VanDerHorn, E.; and Mahadevan, S. 2021. Digital Twin: Generalization, Characterization, and Implementation. *Decision Support Systems*, 145: 113524. doi.org/10.1016/j.dss.2021.113524

Wang, P.; Rowe, J.; Mott, B.; and Lester, J. 2016. Decomposing Drama Management in Educational Interactive Narrative: A Modular Reinforcement Learning Approach. In *Proceedings of the 9th International Conference on Interactive Digital Storytelling*, Los Angeles, CA, USA, 270-282. Springer International Publishing. https://doi.org/10.1007/978-3-319-48279-8_24

Wang, P.; Rowe, J.P.; Min, W.; Mott, B.W.; and Lester, J.C. 2018. High-Fidelity Simulated Players for Interactive Narrative Planning. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, 18: 13-19.

Wiggins, J.B.; Boyer, K.E.; Baikadi, A.; Ezen-Can, A.; Grafsgaard, J.F.; Ha, E.Y.; Lester, J.C.; Mitchell, C.M.; and Wiebe, E.N. 2015. JavaTutor: An Intelligent Tutoring System That Adapts to Cognitive and Affective States During Computer Programming." In *Proceedings of the 46th ACM Technical Symposium on Computer Science Education*, 599. https://doi.org/10.1145/2676723.2691877

Westera, W. 2019. Why and How Serious Games Can Become Far More Effective: Accommodating Productive Learning Experiences, Learner Motivation and the Monitoring of Learning Gains. *Journal of Educational Technology and Society*, 22(1): 59-69.

Woolf, B.P. 2010. *Building Intelligent Interactive Tutors: Student-Centered Strategies for Revolutionizing E-Learning*. Massachusetts: Morgan Kaufmann.

Zhang, X.; Shang, Y.; Ren, Y.; and Liang, K. 2023. Dynamic Multi-Objective Sequence-Wise Recommendation Framework via Deep Reinforcement Learning. Complex and Intelligent Systems, 9(2): 1891-1911. https://doi.org/10.1007/s40747-022-00871-x

Zhou, G.; Azizsoltani, H.; Ausin, M.S.; Barnes, T.; and Chi, M. 2022. Leveraging Granularity: Hierarchical Reinforcement Learning for Pedagogical Policy Induction. *International Journal of Artificial Intelligence in Education*, 32(2): 454-500.