International Journal of Computational Research in Science and Technology
E-ISSN: 3108-0871, Volume 2 Issue 3
May-Jun 2026, PP 01-12

An AI-Driven Personalized Learning Pace Optimizer Using Reinforcement Learning and Self-Paced Curriculum Design

Sara Saroj Pathan[1], Mrunal Jitendra Palaskar[2], Chetan Hiraman Satpute[3], S. N. Jadhav[4]
Department of Artificial Intelligence and Data Science MET Bhujbal Knowledge City, Institute of Engineering Nashik, India

Abstract. The rapid expansion of e-learning platforms has enabled large-scale access to education; however, most existing systems continue to employ static content pacing strategies that fail to accommodate individual learner differences. Such one- size-fits-all approaches often result in learner disengagement, inefficient knowledge acquisition, and high dropout rates. This work presents an AI-driven personalized learning pace optimizer that integrates Reinforcement Learning (RL) with Self-Paced Learning (SPL) to dynamically adapt instructional pacing based on a learner's evolving knowledge state. SPL is used to structure educational content from easy to hard, providing pedagogical stability and robustness to noisy learner data, while RL models the pacing decision as a sequential optimization problem. A multi-objective reward formulation is adopted to balance learner engagement, knowledge retention, and learning efficiency. The proposed approach provides a technically robust and pedagogi- cally safe architecture for adaptive e-learning systems and serves as a foundation for future real-world deployment and evaluation.

Keywords: Personalized Learning, Reinforcement Learn- ing, Self-Paced Learning, Adaptive E-Learning, Curriculum Learning, and Knowledge Tracing.

I. Introduction

The widespread adoption of digital education platforms has fundamentally transformed how learning resources are delivered and consumed. Online learning management sys- tems, Massive Open Online Courses (MOOCs), and intelligent tutoring platforms now support millions of learners worldwide. Despite these advances, a critical limitation persists: most e-learning systems continue to rely on static curricula and fixed pacing mechanisms. These systems assume that all learners progress through educational material at a uniform rate, an assumption that is fundamentally incompatible with the diverse cognitive abilities, prior knowledge levels, and learning preferences of real-world learners.

Learners vary in pace, knowledge, and engagement. Fixed pacing leads to either boredom or cognitive overload, reducing learning effectiveness.

Reinforcement Learning (RL) offers a principled framework for sequential decision-making, where an agent learns an op- timal policy by interacting with an environment and receiving feedback in the form of rewards. In an educational context, RL can be used to determine when to advance a learner to more complex material, when to pro-

vide revision, and when to rein- force foundational concepts. However, naive application of RL introduces significant risks. Excessive exploration may expose learners to content that is too difficult, leading to frustration and disengagement, while poorly designed reward functions may encourage short-term performance at the expense of long- term mastery [6], [7].

Self-Paced Learning (SPL) addresses complementary chal- lenges by formalizing the principle of learning from easy to hard. Inspired by human learning behavior, SPL prioritizes simpler concepts in early stages and gradually incorporates more complex material as learner confidence increases. SPL has been shown to be robust to noisy data and unstable learn- ing signals, making it particularly well-suited for educational environments where learner responses may include guesses, slips, or inconsistent performance [1], [3].

This paper integrates Reinforcement Learning with Self- Paced Learning to design an AI-driven personalized learning pace optimizer that is both adaptive and pedagogically safe. By constraining RL decision-making within an SPL-structured curriculum, the proposed framework balances personalization, learning efficiency, and learner well-being. The main contri- butions of this paper are summarized as follows:

- A unified framework that combines Reinforcement Learn- ing and Self-Paced Learning for adaptive pacing in e-learning systems.
- A formal formulation of learning pace optimization as a sequential decision-making problem.
- A curriculum-constrained RL policy that reduces unsafe exploration and improves interpretability.
- A multi-objective reward design that balances engage- ment, retention, and learning efficiency.

II. Related Work

Adaptive E-Learning Systems

Early adaptive e-learning systems primarily relied on rule- based mechanisms and expert-defined heuristics. These sys- tems adjusted content difficulty or sequence based on pre- defined thresholds such as quiz scores or completion times. While effective in small-scale deployments, rule-based ap- proaches lack flexibility and scalability.

Intelligent Tutoring Systems (ITS) introduced more sophis- ticated learner modeling techniques, incorporating cognitive theories and domain knowledge to personalize instruction. Al- though ITS demonstrated improved learning outcomes, many systems depend on handcrafted models and struggle with uncertainty, noisy data, and large-scale deployment.

Reinforcement Learning in Education

Recent research has explored the application of Rein- forcement Learning to educational problems such as skill mastery sequencing, content recommendation, and tutoring policy optimization. However, educational environments pose unique challenges for RL, including sparse feedback, delayed learning outcomes, and ethical constraints

on exploration. Unconstrained exploration can negatively impact learner ex- perience, highlighting the need for safer and more structured RL formulations.

Curriculum Learning and Self-Paced Learning
Curriculum Learning formalizes the idea of presenting train- ing samples in a meaningful order, typically from easy to hard, to improve convergence and generalization. Self-Paced Learning extends this idea by introducing adaptive mecha- nisms that automatically select appropriate samples based on model confidence. SPL has demonstrated robustness to noisy and ambiguous data and has been applied successfully in various machine learning domains. However, SPL alone does not address online decision-making or personalized pacing.

Research Gap
Existing adaptive learning approaches either rely on static curricula with limited personalization or employ RL with- out sufficient pedagogical constraints. Few systems explicitly integrate curriculum design with sequential decision-making. This gap motivates the proposed RL–SPL framework, which combines structured curriculum progression with adaptive policy optimization to achieve safe and effective personalized learning.

III. Problem Formulation

Personalized learning pace optimization can be formulated as a sequential decision-making problem, where the objective is to determine the most appropriate instructional action for a learner at each interaction step. The system must dynami- cally adapt content difficulty and progression speed based on incomplete and noisy observations of learner behavior.

Let $L = \{l1, l2, \ldots, lN\}$ denote the set of learners interact- ing with the e-learning platform, and let $C = \{c1, c2, \ldots, cM\}$denote the set of instructional content items. Each content item ci is associated with a latent difficulty level d(ci) and a target skill or concept.

Learner State Representation
The learner's true knowledge state is not directly observable and must be inferred from interaction data. Let $st \in S$ represent the learner's latent knowledge state at time step t. This state may be modeled as a multidimensional vector capturing mastery probabilities over a set of skills or concepts. Observable signals include correctness of responses, number of attempts, response time, and interaction frequency.

Due to guessing behavior, careless mistakes, and external distractions, these observations are inherently noisy. As a re- sult, the learning environment is partially observable, requiring the system to make pacing decisions under uncertainty.

Action Space and Pedagogical Constraints
At each interaction step, the system selects an instructional action $at \in A$. In the proposed framework, the action space is deliberately constrained to ensure pedagogical safety. Rather than allowing arbitrary content selection, actions are defined at the curriculum level as follows:

- **Stay:** Maintain the current difficulty level and provide additional practice.
- **Advance:** Progress the learner to more challenging con- tent.
- **Revise:** Revisit previously covered concepts for reinforcement.

These actions operate within the bounds defined by the Self-Paced Learning module, which filters content based on learner readiness. This constrained action space reduces the risk of exposing learners to excessively difficult material and improves interpretability of system decisions.

Reward Modeling

The reward signal guides the optimization of the pacing policy. Unlike traditional systems that rely solely on immediate correctness, educational outcomes require a more nuanced reward structure. Let rt denote the reward received at time step t, which captures multiple learning objectives.

The reward incorporates indicators such as task success, engagement duration, and indicators of knowledge retention. Immediate rewards may reflect correct responses or sustained interaction, while delayed rewards capture long-term mastery and performance on future related tasks. This design discourages short-term optimization strategies that may harm durable learning.

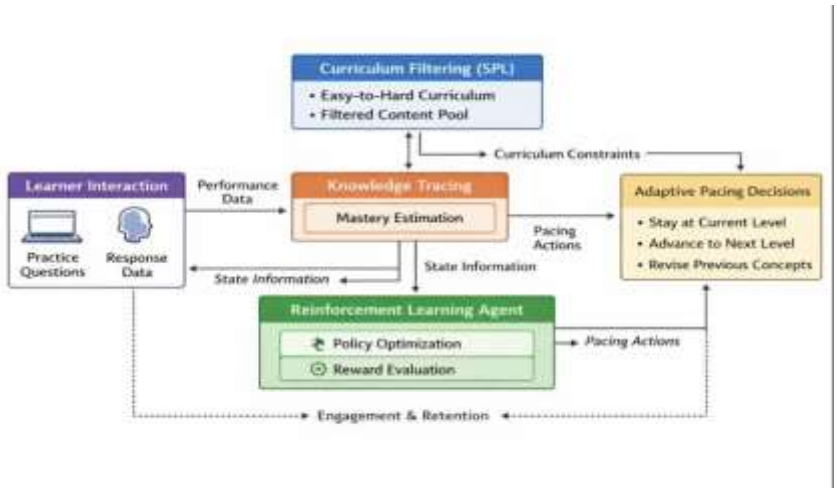

Fig. 1. System architecture of the proposed AI-driven personalized learning pace optimizer

Optimization Objective

The overall objective is to learn a pacing policy π that maximizes the expected cumulative discounted reward:

- **Learner Interaction Layer**

The learner interaction layer serves as the primary inter- face between the learner and the adaptive learning system. Learners engage with instructional content through activities such as solving problems, watching instructional videos, and completing assessments. During these interactions, the system collects rich behavioral data, including response correctness, number of attempts, response time, and session duration.

- **Knowledge Tracing Module**

The knowledge tracing module is responsible for estimat- ing the learner's latent knowledge state based on observed interaction data. This module employs probabilistic or neural knowledge tracing techniques to infer mastery levels over a set of predefined skills or concepts. The output of this module is a continuous-valued state representation that reflects the learner's current level of understanding.

The system captures learning progression over time rather than relying on isolated performance indicators. This represen- tation forms the state input for both the self-paced curriculum module and the reinforcement learning agent [5].

$$\max_{\Pi} E \sum_{t=0}^{T} \gamma r_t \qquad (1)$$

Self-Paced Curriculum Filter

where $\gamma \in [0, 1]$ is a discount factor that balances immediate performance against long-term learning outcomes.

A higher value of γ emphasizes long-term retention and mastery, while lower values prioritize immediate task success. Selecting an appropriate discount factor is critical in educational settings, where the ultimate goal is durable learning rather than short-term accuracy.

Design Challenges

Several challenges arise in optimizing personalized learning pace. First, the partial observability of learner knowledge introduces uncertainty that complicates decision-making. Second, educational rewards are often delayed, making policy learning more difficult. Third, excessive exploration can negatively impact learner motivation and confidence.

The proposed framework addresses these challenges by integrating curriculum constraints through Self-Paced Learn- ing and by designing reward functions that balance multiple educational objectives. Together, these design choices enable safer and more stable optimization of personalized learning pace.

IV. System Architecture

The proposed personalized learning pace optimizer is de- signed as a modular and extensible architecture that integrates learner modeling, curriculum structuring, and decision-making components into a closed feedback loop. The architecture is intentionally decomposed into distinct modules to enhance interpretability, scalability, and pedagogical safety.

Fig. 1 illustrates the overall system architecture, highlighting the interaction between the learner, the knowledge tracing module, the self-paced curriculum filter, and the reinforcement learning–based pace optimizer.

The self-paced curriculum filter structures the entire content repository into an ordered curriculum based on estimated difficulty and learner readiness. Inspired by the principle of learning from easy to hard, this module dynamically selects a subset of content that aligns with the learner's current competence.

Rather than presenting all available content to the decision- making agent, the curriculum filter constrains the action space by excluding content that is either too simple or excessively difficult. This filtering mechanism mitigates the impact of noisy learner responses and prevents abrupt difficulty jumps that could negatively affect learner motivation.

Reinforcement Learning–Based Pace Optimizer

The reinforcement learning–based pace optimizer operates on top of the structured curriculum provided by the self-paced learning module. The RL agent observes the learner's current knowledge state and selects a pacing action, such as maintain- ing the current difficulty level, advancing to more challenging material, or revisiting previously covered concepts.

By restricting the agent's decisions to curriculum-level ac- tions, the architecture significantly reduces unsafe exploration. The agent learns a pacing policy that optimizes long-term ed- ucational objectives while respecting pedagogical constraints imposed by the curriculum structure.

Feedback Loop and Policy Update

After an instructional action is executed, the learner interacts with the selected content, generating new behavioral data. This data is fed back into the knowledge tracing module, updating the learner's estimated knowledge state. The reinforcement learning agent receives a reward signal reflecting learning outcomes, engagement, and retention, which is used to update the pacing policy.

This closed-loop feedback mechanism enables continuous adaptation of instructional pacing. Over time, the system learns personalized pacing strategies that are tailored to individual learners while maintaining pedagogical consistency across the curriculum.

Architecture Diagram Explanation

As shown in Fig. 1, the architecture follows a left-to-right information flow. Learner interactions generate observable signals that are processed by the knowledge tracing module to infer the learner's latent state. This inferred state is simultaneously consumed by the self-paced curriculum filter and the reinforcement learning agent.

The curriculum filter constrains the set of feasible instructional actions, ensuring that the RL agent operates within a pedagogically valid region of the content space. The RL agent then selects a pacing decision, which determines the next learning activity presented to the learner. The resulting learner response closes the feedback loop, enabling continuous policy refinement.

Architectural Advantages

The proposed architecture offers several advantages over existing adaptive learning systems. First, modular separation improves interpretability, allowing educators to

inspect curriculum progression independently of policy optimization. Second, curriculum constraints reduce exploration risk, en- hancing learner safety and trust. Third, the architecture is extensible, enabling future integration of affective comput- ing, context-aware policies, and explainable decision-making mechanisms.

V. Reinforcement Learning Model

To enable adaptive and personalized pacing decisions, the proposed system models the learning process as a sequential

Although the true transition dynamics are unknown, the RL agent learns an optimal policy through repeated interaction with the learning environment.

State Representation
The state st captures the learner's estimated mastery over a set of skills or concepts at time t. This representation may be implemented as a vector of continuous values, where each dimension corresponds to the probability of mastery of a specific concept. Additional contextual features, such as recent performance trends and interaction frequency, may also be incorporated.

Due to partial observability and noise in learner behavior, the estimated state is inherently uncertain. Nevertheless, by updating the state representation sequentially, the agent can capture learning progress over time.

Action Space Design
The action space is intentionally constrained to ensure pedagogical safety and interpretability. At each time step, the agent selects one of the following actions:

- **Stay:** Continue at the current difficulty level with additional practice.
- **Advance:** Move the learner to more challenging material.
- **Revise:** Reinforce previously covered concepts.

These actions are executed within the content subset defined by the self-paced curriculum filter, preventing abrupt or unsafe difficulty transitions.

Multi-Objective Reward Function
Designing an appropriate reward function is one of the most challenging aspects of applying reinforcement learning in education. Optimizing solely for immediate correctness may encourage the agent to present only easy content, undermining long-term learning. To address this issue, a multi-objective reward formulation is adopted.
The reward at time step t is defined as: decision-making problem using Reinforcement Learning (RL). The RL agent interacts with a learner over time, observes

$$r_t = \alpha r_t^{\text{correct}} + \beta r_t^{\text{engage}} + \delta r_t^{\text{retain}} \qquad (2)$$

E. Policy Optimization
The objective of the reinforcement learning agent is to learn a policy π(a|s) that maximizes the expected cumulative discounted reward:

knowledge state.
- A denotes the action space corresponding to pacing decisions.

$J(\pi) = E\pi$

T

t=0

γtrt#

(3)

- P(st+1|st, at) represents the state transition dynamics
induced by learner interaction.
- R(st, at) defines the reward function.
- $\gamma \in [0, 1]$ is the discount factor.

where $\gamma \in [0, 1]$ is the discount factor.
Value-based methods such as Q-learning and policy-based methods such as Actor–Critic architectures are suitable for this problem. Actor–Critic methods are particularly attractive

due to their stability and ability to handle continuous state representations. The actor network proposes pacing actions, while the critic evaluates the expected return of the current policy.

F. Learning Algorithm

Algorithm 1 RL-Based Personalized Learning Pace Optimizer Initialize policy parameters θ and value parameters ϕ
for each learner episode do
Initialize learner state s0
for each interaction step t do Observe current state st Select action at ∼ πθ(a|st)
Execute at under SPL curriculum constraints Observe reward rt and next state st+1
Update critic parameters ϕ
Update actor parameters θ
end for
end for

G. Benefits of Curriculum-Constrained RL
By constraining the RL agent's action space using a self- paced curriculum, the proposed framework significantly re- duces exploration risk and improves learning stability. The agent is prevented from selecting pedagogically inappropriate actions, ensuring that personalization does not come at the expense of learner well-being.
Furthermore, the constrained policy improves interpretabil- ity, as pacing decisions can be explained in terms of curriculum progression rather than opaque content-level choices.

VI. SELF-PACED LEARNING FRAMEWORK

Self-Paced Learning (SPL) provides a principled curriculum design mechanism inspired by the human learning process, where simpler concepts are learned before progressing to more complex ones.

A. Formal SPL Objective

Given a set of training samples {(xi, yi)}N , where xi represents a learning item and yi denotes the learner response, SPL jointly optimizes model parameters and sample selection variables. The objective function is defined as:

B. Advantages of SPL in Educational Settings

In adaptive learning environments, SPL offers several ad- vantages. First, it improves robustness to noisy learner behav- ior by initially focusing on samples that the learner is most likely to answer correctly. Second, SPL stabilizes learning by avoiding abrupt increases in difficulty that may overwhelm learners. Third, SPL provides a transparent curriculum struc- ture that aligns well with pedagogical principles, making system behavior easier to interpret and validate.

C. Integration of SPL with Reinforcement Learning

In the proposed framework, SPL and RL serve complemen- tary roles. SPL is responsible for curriculum structuring, while RL optimizes pacing decisions within that structure. Rather than allowing the RL agent to select any content item, SPL filters the content repository to generate a feasible subset that matches the learner's current competence.

This integration effectively constrains the RL action space, reducing the complexity of policy learning and minimizing unsafe exploration. The RL agent operates at the curriculum level, selecting actions such as staying at the current difficulty, advancing to harder content, or revising previous concepts, while SPL ensures that the selected content remains pedagog- ically appropriate.

D. Curriculum-Constrained Decision Process

At each interaction step, the learner's knowledge state estimated by the knowledge tracing module is provided to the SPL component. Based on this state, SPL determines a difficulty range that the learner is ready to engage with. The RL agent then selects a pacing action that operates within this range.

This two-stage decision process decouples curriculum de- sign from policy optimization, resulting in improved stabil- ity and interpretability. Educators can independently inspect curriculum progression rules without delving into RL policy parameters.

E. Use-Case Walkthrough

To illustrate the behavior of the proposed framework, consider a novice learner beginning an introductory algebra course. Initially, the learner's estimated mastery is low across all concepts. The SPL module restricts content selection to low-difficulty items, such as basic arithmetic operations.

As the learner interacts with the system, the knowledge tracing module updates mastery estimates based on perfor- mance trends rather than isolated responses. Occasional incor- rect answers do not immediately trigger regression, as SPL

N N filtering mitigates the impact of noisy data. When consistent

$$\min \Sigma v_i L(y_i, f(x_i, w)) - \lambda \Sigma v_i \quad (4)$$

improvement is detected, the RL agent selects the Advance

w,v
i=1

i=1

action, progressing the learner to more challenging content such as linear equations.

where w denotes model parameters, $v_i \in [0, 1]$ indicates
whether sample i is included in training, $L(\cdot)$ is a loss function, and λ is the pace parameter controlling learning progression.
As λ increases over time, progressively harder samples are incorporated into the training process.

If the learner exhibits difficulty or disengagement at a higher difficulty level, the RL agent may select the Revise action, reinforcing prerequisite concepts. This adaptive behavior con- tinues throughout the learning session, enabling personalized pacing that balances challenge and support.

F. Educational Impact

Compared to static pacing strategies, the proposed RL–SPL framework dynamically adapts to individual learner needs. By integrating curriculum design with sequential decision- making, the system promotes sustained engagement, reduces cognitive overload, and supports durable learning. These prop- erties are essential for scalable and effective digital education platforms.

VII. DISCUSSION

The proposed RL-SPL framework demonstrates how cur- riculum design and sequential decision-making can be ef- fectively combined to address the limitations of static e- learning systems. By grounding reinforcement learning deci- sions within a self-paced curriculum, the framework ensures pedagogical safety while maintaining adaptability to individual learner needs.

A key advantage of the proposed approach is its modular architecture. Learner modeling, curriculum structuring, and policy optimization are treated as independent yet intercon- nected components. This separation of concerns improves system interpretability and allows educators to reason about curriculum progression without requiring detailed knowledge of reinforcement learning algorithms.

Furthermore, the use of a multi-objective reward function enables the system to balance competing educational goals. Rather than optimizing solely for immediate correctness, the framework explicitly incorporates engagement and retention, which are critical for long-term learning success. This design choice mitigates the risk of degenerate policies that prioritize short-term gains at the expense of durable knowledge acqui- sition.

A. Limitation

Despite its advantages, the proposed framework has several limitations. First, the cold-start problem remains a challenge, as limited data is available when a learner

initially joins the system. Although curriculum constraints mitigate risk, early pacing decisions may still be suboptimal until sufficient interaction data is collected.

Second, accurate learner modeling is critical for effec- tive adaptation. Errors in knowledge tracing may propagate through the decision-making pipeline, affecting pacing de- cisions. While SPL improves robustness to noise, further advances in learner modeling are required to fully address this issue.

Third, computational complexity may pose challenges in large-scale deployments involving thousands of concurrent learners. Efficient policy learning and scalable state represen- tations are essential for real-time personalization.

B. Ethical and Privacy Considerations

Ethical considerations play a crucial role in adaptive edu- cational systems. The collection and processing of learner in- teraction data raise important privacy concerns. Systems must

adhere to data protection regulations and ensure transparency in how learner data is used.

Explainability is another important ethical requirement. Learners and instructors should be able to understand why certain pacing decisions are made. The curriculum-constrained nature of the RL agent improves explainability compared to unconstrained content recommendation systems.

VIII. CONCLUSION AND FUTURE WORK

This paper presented an AI-driven personalized learning pace optimizer that integrates Reinforcement Learning with Self-Paced Learning to address the limitations of static e- learning systems. By modeling instructional pacing as a sequential decision-making problem and constraining policy optimization within a pedagogically structured curriculum, the proposed framework achieves safe, interpretable, and effective personalization.

The framework balances multiple educational objectives, in- cluding engagement, retention, and learning efficiency, through a multi-objective reward formulation. Its modular design fa- cilitates scalability, interpretability, and future extension.

Future work will focus on real-world deployment and em- pirical evaluation using large-scale learner datasets. Additional directions include integrating affective computing signals to capture learner emotions, developing explainable reinforce- ment learning techniques for educational decision-making, and exploring meta-learning approaches to address the cold-start problem. These advancements will further enhance the effec- tiveness and trustworthiness of AI-powered adaptive learning systems.

REFERENCES

[1] Y. Bengio, J. Louradour, R. Collobert, and J. Weston, ”Curriculum learning,” in Proc. 26th Int. Conf. Machine Learning (ICML), 2009, pp. 41–48.

[2] A. Muppidi, Z. Zhang, and H. Yang, ”Pick up the PACE: A Parameter- Free Optimizer for Lifelong Reinforcement Learning,” arXiv preprint arXiv:2402.06778, 2024.

[3] D. Meng, Q. Zhao, and L. Jiang, ”What Objective Does Self-Paced Learning Indeed Optimize?,” arXiv preprint arXiv:1511.06049, 2016.

[4] H. Li, M. Gong, D. Meng, and Q. Miao, ”Multi-Objective Self-Paced Learning,” in Proc. AAAI Conf. Artificial Intelligence, vol. 34, no. 04, 2020, pp. 4735–4742.

[5] C. Piech, J. Bassen, J. Huang, S. Ganguli, M. Sahami, L. J. Guibas, and

J. Heer, ”Deep knowledge tracing,” in Advances in Neural Information Processing Systems (NeurIPS), 2015, pp. 505–513.
[6] R. S. Sutton and A. G. Barto, Reinforcement learning: An introduction.
MIT press, 2018.
[7] V. Mnih et al., ”Human-level control through deep reinforcement learn- ing,” Nature, vol. 518, no. 7540, pp. 529–533, 2015.
[8] V. R. Konda and J. N. Tsitsiklis ”Actor-critic algorithms,” in Advances in Neural Information Processing Systems (NeurIPS), 2000, pp. 1008–1014.
[9] M. P. Kumar, B. Packer, and D. Koller, ”Self-paced learning for latent variable models,” in Advances in Neural Information Processing Systems (NeurIPS), 2010, pp. 1189–1197.
[10] S. D’Mello and A. Graesser, ”Affective computing and automated adapters in advanced learning technologies,” in Emotions, technology, and learning, 2015, pp. 183–205.
[11] D. Doran and G. L. T. Parra, ”Contextual-bandits for personalized learning,” in Proc. 20th Int. Conf. Artificial Intelligence in Education (AIED), 2019, pp. 126–137.