Emotions as implicit feedback for adapting difficulty in tutoring systems based on reinforcement learning

Abstract

In tutoring systems, a pedagogical policy, which decides the next action for the tutor to take, is important because it determines how well students will learn. An effective pedagogical policy must adapt its actions according to the student's features, such as knowledge, error patterns, and emotions. For adapting difficulty, it is common to consider student knowledge but not the other features as emotions. Reinforcement learning (RL), which is a machine learning framework, fits well for adapting to difficulty; however, the known ways of considering emotions into RL like through states or reward-shaping functions are not enough. Then, to find the pedagogical policy that maximizes the student learning gain, we propose considering emotions as implicit feedback through both the reward and the exploration-exploitation strategy, using the circumplex model to represent emotions and the flow theory to select the appropriate difficulty level. Our approach follows three design considerations: pursuing positive emotions, managing unwanted (anxiety and boredom) emotions, and anticipating unwanted emotions. We simulate interactions with users based on real data from publicly available datasets to quantitatively compare our approach with others that adapt difficulty. Also, we qualitatively compare our approach with others that consider emotions in different contexts. Quantitative results show that our approach is better than the others that adapt difficulty to foster learning gain in students because it allows getting higher values all the studied time (200 tasks). Qualitative comparisons show that although other approaches pursue positive emotions or manage unwanted emotions, our approach does so as well and additionally anticipates unwanted emotions. We conclude that our approach is useful in tutoring systems for adapting difficulty because it allows high learning gains in students in a few interactions.

Keywords. Emotions, adapting difficulty, tutoring systems, reinforcement learning.

Statements and Declarations

Competing Interests

Not Applicable.

1. Introduction

In domains like math, probability, and logic, solving a problem often requires producing an argument, proof, or derivation consisting of one or more inference "steps" (Zhou et al., 2022). In such domains, tutoring can be described as a two-loop procedure (Vanlehn, 2006). The outer loop governs problem-level pedagogical decisions such as selecting the next problem or task for the student to work on. The inner loop controls step-level pedagogical decisions such as whether to give feedback or to prompt the student with an example. In this context, a pedagogical policy is used to decide the next action for the tutor to take among a set of alternatives (Shen, 2018). This process is challenging because, on the one hand, how instruction is sequenced can make a difference in how well students will learn (Ritter et al., 2007); on the other hand, each decision of the pedagogical policy affects the student's subsequent actions and performance, which also has an impact on the tutoring system's next decision (Ausin, 2019). An effective tutoring system would craft and adapt its actions to the student's needs (Chi et al., 2010). In general, a tutoring system tends to adapt its behavior considering one of five features (Alaven et al., 2017): knowledge, error patterns, self-regulation of learning, learning styles, and emotions.

Adapting difficulty, which is the interest of this paper, is relevant in different contexts like adaptive training (Fraulini et al., 2023), cognitive training (Zini et al., 2022), video games (Sepulveda et al., 2019), serious games (Seyderhelm & Blackmore, 2021), among other domains. Particularly, in tutoring systems, a challenge is to suit the difficulty level of tasks to the current student's skills because the system should not provide those too easy and leave the student bored or too hard to the point that they discourage the student. Reinforcement learning (RL) is a machine learning technique in which an agent learns what to do through trial-and-error interactions with an environment to achieve a goal, making it a very useful approach for adapting a system to new contexts. Azoulay et al. (2014) present a comparison of available algorithms to adapt difficulty in tutoring systems, where most of them are based on RL, such as Q-learning, Virtual Learning, and Deviated Virtual Reinforcement Learning (DVRL). For that purpose, the Q-learning algorithm assumes that only one state exists, where the actions indicate the different difficulty levels, the reward is related to the success in answering, and only one Q value is updated from an interaction. Virtual Learning and DVRL are variations in which more than one Q value is updated from an interaction. All those algorithms were compared in simulations according to the agent learning point of view and all of them adapt difficulty only from the student knowledge. In this paper, we focus on covering the lack of comparisons according to the student learning point of view and we explore another feature of adaptation, which is to consider student emotions as well rather than only knowledge.

Emotions can be useful to know if the student is discouraged by the material or disengages from the tutoring system (Gordon et al., 2016). Particularly, tutoring systems have shown learning gains in students when they consider students' emotions by applying hand-coded rules to determine their actions (D'Mello et al., 2010; D'Mello et al., 2012; Salazar et al., 2021). However, given the continuously evolving interactions where user needs and preferences change over time, hand-coded rules are labor-intensive (Akalin & Loutfi, 2021), which makes it difficult to create rules in real time. That challenge is addressed by RL-induced pedagogical policies (Zhou et al., 2022), which are policies automatically learned from interactions according to the RL framework. RL-induced pedagogical policies that consider the student's emotions are based on SARSA (Gordon et al., 2016), Q-learning (Park et al., 2019; Pérez et al., 2023), or MAXQ (Chan & Nejat, 2012). Gordon et al. (2016) consider student emotions as three discrete states (neg, med, and pos) that represent negative, medium, and positive values of emotional valence, to determine what emotion must be expressed in a tutoring system. Park et al. (2019) also define emotions as discrete states (q1, q2, q3, and q4) that represent quartiles of emotional valence to adapt sentences for storytelling. Pérez et al. (2023) use continuous values, gotten from emotional valence and arousal, as a reward-shaping function to adapt the topic in sessions of training math word problems. Chan & Nejat (2012) consider emotions as states of four categories (stressed, neutral, excited, and pleased) to adapt behaviors of a social robot in a memory game. To our knowledge, there is not an approach that includes students' emotions to adapt difficulty.

Although students' emotions are incorporated as states (Chan & Nejat, 2012; Gordon et al., 2016; Park et al., 2019), or as a reward-shaping function (Pérez et al., 2023), it is necessary to explore another way because our case study does not fit well in any of them. On the one hand, incorporating emotions as states means either transforming dimensional values into categories like Gordon et al. (2016) or directly using categories like Chan & Nejat (2012), which is wasting useful information, for example, how far the current emotion of target emotions is. On the other hand, incorporating emotions as a reward-shaping function does not allow managing unwanted emotions such as selecting a lower difficulty level when the student is frustrated. Then, in this paper, we propose including emotions as implicit feedback for adapting difficulty by using an algorithm based on RL as well but considering emotions in both the reward and the exploration-exploitation strategy, which is a balance between trying new actions (exploration) and using the best policy that the agent has identified so far (exploitation). Specifically, the goal of our approach is to find the pedagogical policy that maximizes student learning by selecting tasks with 5 skills with different levels of difficulty, according to the student's response and emotional expression while solving the tasks.

Particularly, we propose an approach that represents emotions using the circumplex model of emotions, and relates them to students' abilities and difficulty levels using flow theory. Subsequently, it uses an RL algorithm to define the human feedback process in the tutoring system, to adapt the difficulty. Our main contributions are:

1) An algorithm based on RL for personalized pedagogical policies based on adapting difficulty.
2) A novel approach of incorporating user emotion in emotion-aware RL-based algorithms.
3) A quantitative comparison of available algorithms to adapt difficulty in tutoring systems.
4) A qualitative comparison of available emotion-aware reinforcement learning-based algorithms.

This paper is organized as follows: in Section 2, we present the main concepts related to our approach; Section 3 describes our approach; in Section 4, we describe the experimental protocols; in Section 5, we present quantitative results; in Section 6, we qualitatively compare our approach with related works. Finally, in Section 7, we present the conclusions and future works.

2. Theoretical framework

Our approach is based on the implicit human feedback from the student's emotional expression to select the appropriate level of difficulty using a reinforcement learning algorithm. Section 2.1 presents how emotions are represented, and particularly, why we select the circumplex model of emotions, and, Section 2.2 describes a framework, called flow theory, which relates emotions, student skills, and levels of difficulty. Later, Section 2.3 points out the fundamentals of reinforcement learning through human feedback, and subsequently, we describe the algorithms used in tutoring systems to adapt to difficulty in Section 2.4.

2.1. Emotions in tutoring systems

Human emotions, among other things, are part of human communication (Pérez et al., 2018), so, it is necessary to interpret emotions to understand better what a person is trying to communicate. In academic contexts, students' learning is related to four groups of emotions (Johri, 2023): achievement, epistemic, topic, and social emotions. Achievement emotions are related to success and failure resulting from achievement activities, such as happiness for finishing an activity. Epistemic emotions are triggered by cognitive problems, such as surprise about a new task; topic emotions are related to the theme presented in lessons, where both positive and negative emotions can trigger students' interest in learning material. Social emotions are triggered by interacting with teachers and peers. Although it is difficult to exactly determine what is the trigger of emotions in the academic domain, we argue that emotions are there and can be measured while solving a task.

In tutoring systems, which are a kind of Human-Computer Interaction (HCI), the emotional states of a user are incorporated into the decision cycle of the interface to develop more influential, friendly, and natural applications, which is known as Affective Computing (Wang et al., 2022). Emotions are so important for humans that, for example, machines expressing emotions improve the HCI (Pérez et al., 2020). According to Landowska (2018), an analysis reveals that there is no one commonly accepted standard model for emotion representation, but it can be categorized at least into three types: discrete, dimensional, and componential. Discrete models distinguish a set of basic emotions, such as Ekman's six basic emotions model that includes joy, anger, disgust, surprise, sadness, and fear (Ekman & Friesen, 1971), or simple models of three categories as joy, neutral, and sad (Pérez & Castro, 2018). Dimensional models represent an emotional state as a point in a multi-dimensional space (Salazar et al., 2021), such as the circumplex model that represents emotions as a point in a space of two continuous dimensions of valence and arousal (Russell, 1980). Finally, componential models consider several factors that influence the resulting emotional state, such as the OCC model that defines a hierarchy of 22 emotion types (Ortony et al., 1988).

For our approach, we select the circumplex model because it allows us to pursue a pedagogical policy based on emotions by optimizing the valence and interpreting both valence and arousal. Fig. 1 shows a representation of the circumplex model of emotions, in which valence represents the horizontal axis and arousal represents the vertical axis. The model divides emotions into four quadrants: the top right quadrant contains emotions that are high in valences and high in arousal, such as excited and happy; the bottom right quadrant contains emotions that are high in valence but low in arousal, such as calm and relaxed; bottom left quadrant contains emotions that are low in valence and low in arousal, such as sad and bored; top left quadrant contains emotions that are low in valence but high in arousal, such as angry and distressed. Each emotion in the circumplex model has a degree value, for example, excited, frustrated, and bored, are located at 48.6, 141, and 242 degrees, respectively.

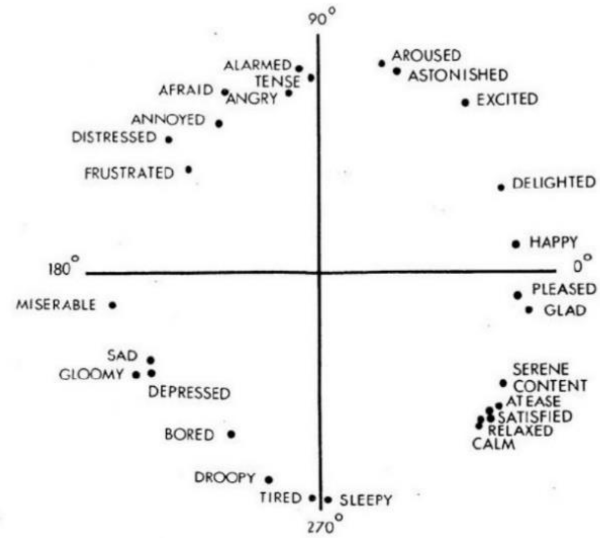

Fig. 1. Circumplex model (Russell, 1980)

User emotions can be obtained manually by asking the user or an observer, and automatically by extracting those from user expressions in different modes (Wang et al., 2022): language, voice, face, body posture, physiological signals, among others. Considering that facial expression is always available, in our approach, we propose to use it to capture emotions from spontaneous reactionsSeveral tools like Affdex (McDuff et al., 2016) allow getting the emotion from the facial expression. That is, given a picture with the facial expression, the tool returns an emotional value. In our case, in which we wanted to relate emotions with solving a task, we propose to get the emotional value each second and calculate the average value from the beginning to the end of the task.

2.2. Flow theory

The flow theory is a symbiotic relationship between challenges and skills needed to meet those challenges, where the flow is believed to occur when one's skills are neither overmatched nor underutilized to meet a given challenge (Shernoff et al., 2003). Flow is a state of deep absorption in an activity that is intrinsically enjoyable, as when artists or athletes are focused on their play or performance (Csikszentmihalyi, 1990). In the context of videogames, a related to our case study and well-known application of the flow theory is to adapt the game's features, behaviors, and scenarios in real-time, depending on the player's skill, so that the player does not feel bored when the game is very simple or frustrated when it is very difficult (Zohaib & Nakanishi, 2018). To be more illustrative, Fig. 2 represents the flow model in the context of videogames, which relates the player's skill and the game challenge: when the difficulty of the game is higher than the player's skill, the activity becomes frustrating, pushing the player into a state of anxiety; when the player skill is higher than the difficulty, then the game is too easy, pushing the player into a state of boredom; when neither of those happens, then the user is faced by a challenge whose difficulty level matches the player's skill, enabling him to enter the flow channel. The main idea is that providing an appropriate series of personalized challenges allows the player to stay in the flow channel for longer periods. For example, the experience of anxiety may prompt the game to decrease the level of the game challenge.

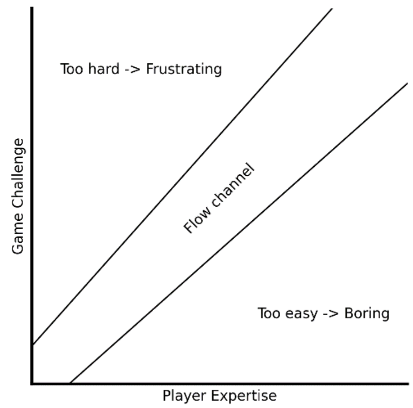

Fig 2. Flow theory in the context of videogames (Zohaib & Nakanishi, 2018)

Similarly, in tutoring systems, problems are often assigned adaptively according to the student's skill levels, and the student is generally expected to flow when learning through problems, which is useful to identify students at risk, or modalities of interaction that lead to optimal and suboptimal learning (Kang et. al, 2024). Then, in our approach, we use the flow theory to guide decision-making. That is, if the user is frustrated or bored, then we try to drastically change his emotion by selecting the simplest or hardest task, respectively. Otherwise, we try to select the task that keeps positive emotions.

2.3. Reinforcement learning and human feedback

Reinforcement learning can be formalized as a Markov Decision Process (MDP) (Van Otterlo & Wiering, 2012), where the agent perceives the states of its environment, takes actions that change the states, and receives rewards according to the states achieved. Formally, an MDP is a tuple $< S, A, T, R >$, in which S is a finite set of states, A is a finite set of actions, T is a transition function defined as $T: S \times A \times S \rightarrow [0,1]$, and R is a reward function defined as $R: S \times A \times S \rightarrow \mathbb{R}$. The typical goal is to learn the optimal policy π^*, or nearly optimal policy, which maximizes the expected cumulative reward. A policy can be deterministic or stochastic. A deterministic policy π is a function defined as $\pi: S \rightarrow A$. A stochastic policy is defined as $\pi: S \times A \rightarrow [0, 1]$, such that for each state $s \in S$ (except terminal states), it holds that $\pi(s, a) \geq 0$ and $\sum_{a \in A} \pi(s, a) = 1$.

To learn the optimal policy, the agent must find a balance in the exploration-exploitation trade-off. Exploration refers to trying new actions to gather data from less known areas of the state-action space, while exploitation refers to using the best policy that the agent has identified so far. It means that the agent must explore the environment by performing actions and perceiving their consequences through rewards, and then, it can exploit this knowledge. According to Sutton & Barto (2018), a common strategy is to apply the $\varepsilon - greedy$ function that performs a random action with probability ε, or an action based on the policy $\pi(s)$ with probability $1 - \varepsilon$. If $\varepsilon = 0$, then the action always is based on the policy, which is called $greedy$. A common algorithm to learn the optimal policy is Q-learning (Watkins & Dayan, 1992), which focuses on estimating incrementally Q-values, which are values that relate to a pair state-action. To estimate the Q-values, Q-learning applies an equation that depends on four values and two hyperparameters (see Equation

1). The four values are: state s, action a, next state s', and reward value $R(s')$. The two hyperparameters are the learning rate $\alpha \in [0,1]$ that determines the update rate, and the discount factor $\gamma \in [0,1]$ that determines the value of future rewards. After training, the optimal policy $\pi^*(\mathrm{s}) = argmax_a \mathrm{Q}^*(\mathrm{s}, \mathrm{a})$ is obtained through the argument of the maxima Q-value, which is the action at that the value is maximized.

$$Q(s,a) = (1-\alpha)Q(s,a) + \alpha[R(s') + \gamma max_{a'} Q(s',a')] \quad (1)$$

On the other hand, in RL, learning from human feedback is known as Interactive Reinforcement Learning (IRL), which treats human feedback as a reinforcement signal after the executed action (Tsiakas et al., 2018). IRL usually is applied to improve the convergence speed because an external trainer usually provides guidance in specific states during the learning process. According to Cruz et al. (2016), there are two main approaches based on how feedback is integrated into the RL framework (Cuartas et al, 2023): reward shaping and policy shaping. In reward shaping, usually, an external trainer evaluates how well or badly performed actions by the agent are. In policy shaping, the action proposed by the agent can be replaced by a more suitable action chosen by the external trainer before it is executed. For adapting difficulty according to human emotions, like in our case, we are interested in incorporating human feedback in the exploration-exploitation strategy.

In general, human feedback can be classified into two groups (Akalin & Loutfi, 2021): explicit feedback, when the feedback is direct, provided through an interface such as ratings and labels; and implicit feedback, if the human feedback is spontaneous behavior or reactions such as non-verbal cues and social signals. The reward strategies are categorized into four groups (Akalin & Loutfi, 2021): reward-focused strategy (positive reward for correct actions and no feedback for incorrect actions), punishment-focused strategy (no feedback for correct actions and punishment for incorrect actions), balanced strategy (positive reward for correct actions and punishment for incorrect actions), and inactive strategy (the human teacher rarely provides feedback). In our case, we use implicit feedback and a balanced strategy. On the one hand, it is implicit feedback because we propose to get emotions from spontaneous facial expressions while the student solves a task. On the other hand, the balanced strategy fits better than the other strategies because the values in the circumplex model of emotions are negatives and positives, and we want to avoid the negatives and pursue the positives.

2.4. Algorithms to adapt difficulty in tutoring systems

In tutoring systems, RL has been used for estimating student proficiency (Pérez et al., 2022) but the more common is for inducing pedagogical policies. Azoulay et al. (2014) present algorithms for selecting the appropriate difficulty level, three based on RL (Q-learning, Virtual learning, DVRL), and Bayesian learning. The next sections explain each one.

2.4.1. Q-learning

It is a derived version of Q-learning in which only one state exists, and the updates of Q values are based on Equation 2 where each level is associated with an action. The idea is that the reward indicates the success or failure in answering a question at that level. It means that $argmax_a Q(a)$ will return the action that maximizes the answer success.

$$Q(a) = Q(a) + \alpha[r + \gamma max_{a'} Q(a') - Q(a)] \quad (2)$$

2.4.2. Virtual learning

It is also like Q-learning but instead of learning only from actions and payoffs experienced, the algorithm can also learn by reasoning from the chosen action for other actions. That is, once a student succeeds in answering a question, the Q value of the current level, as well as the Q value of the lower levels, are increased because it assumes that if a student masters a level, then he/she masters the lower levels as well. Similarly, if a student fails to answer a question, then the Q value of the level of the current question as well as the Q value of the higher levels are reduced. It assumes that a student failing in a specific level will fail in the higher levels.

2.4.3. DVRL

It is like Virtual Learning, but once a reward is received for the student's answer, the updating phase of the Q values relates not only to the given question's level but also to the level of the neighboring questions assuming that the closer levels are very related about mastering them. That is, once a student answers a question correctly, the Q value of the nearest higher level also increases, and when a student fails to answer a question, then the Q value of the nearest lower level also decreases.

2.4.4. Bayesian learning

This algorithm assumes a normal distribution of the student's level. Initially, the algorithm associates a constant probability for each set of parameters μ and σ. In each step, the algorithm considers all possible distributions of the student, and for each question's level, the algorithm calculates the expected utility of this level given all possible distributions of students, and then it chooses the level with the highest expected utility. Once a question is chosen and the student's response is observed, the probability of each distribution of the student is updated using the Bayesian rule shown in Equation 3. Then, Equation 4 determines the next level, where $pWins(l \mid \mu, \sigma)$ is the probability of a question from level l to be chosen, $util(l)$ is the utility of a successful answer to a question from this level, and $utilFail$ is the utility of failure to answer a question from this level.

$$P(\mu, \sigma) = \frac{P(\mu, \sigma) * pWins(l \mid \mu, \sigma)}{sumProb(l)} \quad (3)$$

$$next\ l = argmax_l \sum_{\mu,\sigma} P(\mu, \sigma) * pWins(l \mid \mu, \sigma) * util(l) + (1 - pWins(l \mid \mu, \sigma) * utilFail) \quad (4)$$

As our approach's goal is to select the appropriate difficulty level, we use those algorithms (Q-learning, Virtual learning, DVRL, and Bayesian learning) to compare quantitatively the performance of our approach.

3. Our approach

In the context of tutoring systems, we propose to include emotions according to the circumplex model and flow theory in the RL framework to adapt difficulty. To formalize the problem, let us define 5 skills graded in different difficulty levels (i.e. from 1 to 5) and interactions between a student and the system, where an interaction is represented as a tuple (skill, outcome, emotion), being *skill* the category of the task that the system selects, *outcome* the answer of the student, and *emotion* the expression of the student while solving the task. The objective is to find the policy that maximizes the student learning gain.

Our approach focuses on three design considerations: first, keeping the flow channel that is related to pursuing positive emotions; second, managing anxiety or boredom by reducing or augmenting the difficulty (i.e., when being in any of them); and third, anticipating anxiety or boredom by managing them before being achieved (i.e., when being closed to any of them). The general framework is presented in Fig. 3, in which a tutoring system provides a task (also called action), the user responds deliberately with the solution, and the tutoring system selects the next task according to the spontaneous user's emotions. As emotions can be interpreted as points, for detecting anxiety or boredom (which are discrete emotions), we work with circular areas where any point inside means belonging to it. For example, the circular area of anxiety includes points with negative valence and positive arousal (see A1 in Fig. 4).

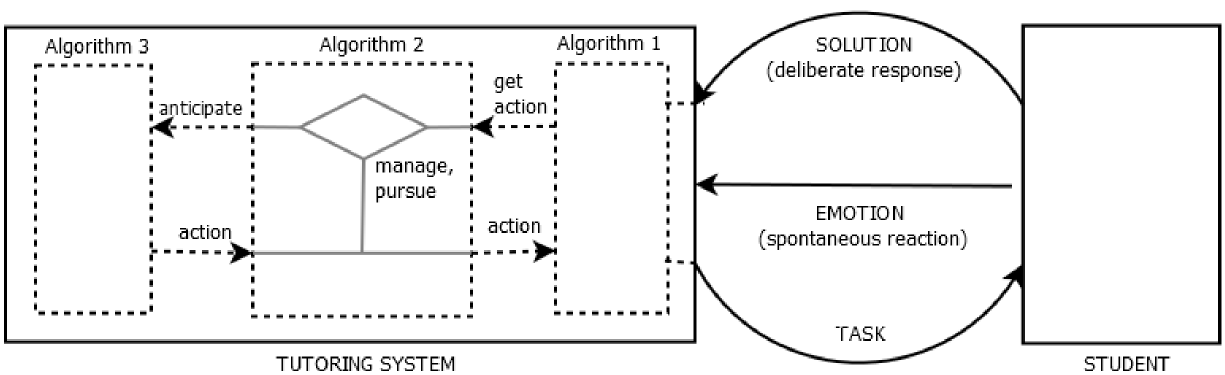

Fig. 3. General framework

The general idea of our method is to adapt an RL algorithm to learn two Q-tables (one that allows relating skills and difficulty levels and the other that relates skills and emotions) such that the first Q-table provides the skill when it is necessary to manage unwanted emotions, and the second one provides the skill that promotes the more positive emotions. Managing unwanted emotions means selecting a skill to drastically change the user's current emotion according to the flow theory (i.e. if it detects anxiety or boredom then the next action will be the easiest or hardest task, respectively). On the other hand, promoting more positive emotions tries to increase the time the learner is in a state of flow.

Specifically, we design an algorithm based on Q-learning inspired by the version presented by Azoulay et al. (2014), where they assume no changes in the state by applying Equation 2 as an update. Our algorithm incorporates three changes that correspond with our three design considerations. First, in the main loop (see Algorithm 1), although we learn the usual $Q_{performance}$

(a Q table to learn the appropriate task difficulty for the user), we learn another Q_{flow} (a Q table to identify the task that better matches the flow channel). Second, in case of exploitation (see Algorithm 2), rather than using always $argmax(Q_{performance})$, we use $argmin(Q_{performance})$ as well when necessary to get the easiest and hardest tasks from the same Q table. Third, in case of exploration for a ε-$greedy$ based strategy (see Algorithm 3), the epsilon value depends on the area that results from intercepting the circular area generated by the current user emotion and the defined circular areas of anxiety or boredom to favor emotion management when the user emotion is closer to them and favor exploration otherwise. Fig. 3 shows how the three algorithms interact with the other components of the framework: Algorithm 1 is the main loop that interacts with the student, Algorithm 2 calculates the action to manage unwanted emotions and pursue positive emotions, and Algorithm 3 calculates the action to anticipate unwanted emotions.

Algorithm 1 shows that we initialize the values of $Q_{performance}$ and Q_{flow} in zero, and $emotion$ randomly. On the one hand, the Q values are zero because our algorithm does not consider any information about the user before starting. On the other hand, emotion is a random value because it will allow getting a random action at the start. Considering that an action is the skill of a task in our context, the loop consists of getting an action according to the current user emotion, performing that action, observing the new user emotion and answer, and updating the Q values. Observing the emotion means calculating the average valence and arousal after getting the values each second from the beginning to the end of the task through facial expressions by using a tool like Affdex (McDuff et al., 2016). Observing the answer means to determine whether the solution is correct or not. For updating $Q_{performance}$, the reward is 1 when the answer is correct but 0 otherwise. It allows knowing the easiest and hardest tasks by applying $argmax(Q_{performance})$ and $argmin(Q_{performance})$, respectively. For updating Q_{flow}, the reward is the valence of the current user emotion to allow us to look for the more positive emotions by using $argmax(Q_{flow})$.

Algorithm 1. Main loop

Initialize $Q_{performance} = 0$
Initialize $Q_{flow} = 0$
Initialize $emotion$ randomly
Loop:
    Get $action$ from $emotion$ using Algorithm 2
    Perform $action$
    Observe $emotion, reward$
    Update $Q_{performance}$ using Equation 2
    Update Q_{flow} using Equation 2

Algorithm 2 shows the decision-making process when anxiety or boredom. From lines 1 to 5, we manage anxiety and boredom by evaluating if the $emotion\_point$ is inside of $anxiety\_area$ or $boredom\_area$ (see example in the second quadrant in Fig. 4 where a point is inside of A1). If it detects anxiety or boredom then the next action will be the $argmax(Q_{performance})$ or $argmin(Q_{performance})$, which are the easiest and hardest tasks, respectively. It tries to drastically change the user's current emotion according to the flow theory. From lines 7 to 9, we anticipate anxiety or boredom by evaluating if the $emotion\_area$ intercepts the $anxiety\_area$ or $boredom\_area$ (see example in the third quadrant in Fig. 4 where a point is inside of B2). We call it anticipate because the current emotion is not inside anxiety (A1) or boredom (B1) areas, but it is close to any of them (see A2 and B2 areas in Fig. 4). Then, if the $emotion\_area$ intercepts any of the others (see how to calculate the intercepted area in the next paragraph), then the next task will be like managing anxiety or boredom but using an ε-$greedy$ strategy to promote exploration. Finally, in line 11, for other cases where anxiety or boredom are not detected or anticipated, we try to keep the flow channel by selecting the task that promotes more positive emotions through $argmax(Q_{flow})$.

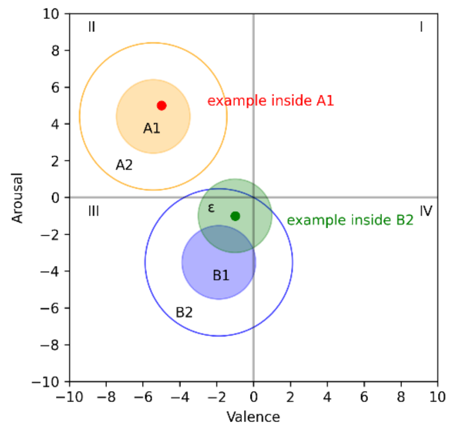

Fig. 4. Example of emotions

Algorithm 2. Manage and pursue emotions

Input: valance, arousal

1. Create $emotion\_point$ from $valence$ and $arousal$
2. If $emotion\_point$ in $anxiety\_area$:
3.   return $argmax(Q_{performance})$
4. Else if $emotion\_point$ in $boredom\_area$:
5.   return $argmin(Q_{performance})$
6.
7. Create $emotion\_area$ from $emotion\_point$ and $radius$
8. If $emotion\_area$ intercepts $anxiety\_area$ or $boredom\_area$
9. $return\ action$ using Algorithm 3
10.
11. $return\ argmax(Q_{flow})$

Output: action

Algorithm 3 shows the ε-$greedy$ strategy to promote exploration while anticipating anxiety or boredom. Having evaluated that the $current\_area$ intercepts another area, which is called $nearest\_area$ (i.e., the current emotion is inside either A2 or B2, so $nearest\_area$ could be anxiety or boredom), we calculate the intercepted area because it will allow us to calculate ε according to that proportion intercepted. Equation 5 calculates de intercepted area, where r is the segment between the current emotion point and the interception point of the circles, R is the segment between the nearest area center and the interception point of the circles, L is the segment between the current emotion point and the nearest area center, α is the angle between r and L, and β is the angle between R and L (see an example in Fig. 5 that presents the interception of boredom and current point, showing the meaning of the parameters). Having the $intercepted\_area$, we proceed to calculate the $intercepted\_proportion$ by dividing $intercepted\_area$ and $nearest\_area$ (see line 2 in Algorithm 3). Because ε represents the probability of exploration, we want it to be low when the emotion is closer to the area (to favor exploitation that means managing emotions) but high otherwise (to favor exploration). We achieve that behavior with $\varepsilon = 1 - intercepted\_proportion$. Finally, if the case is not exploration (determined by lines 4 and 5 in Algorithm 3), according to the intercepted area (anxiety or boredom) will be returned $argmax(Q_{performance})$ or $argmin(Q_{performance})$, respectively.

$$intercepted\_area = \alpha r^2 + \beta R^2 - \frac{1}{2} r^2 \sin 2\alpha - \frac{1}{2} R^2 \sin 2\beta \quad (5)$$

Algorithm 3. Anticipate emotions

Input: $current\_area, nearest\_area$
1. Calculate $intercepted\_area$ using Equation 5
2. $intercepted\_proportion = intercepted\_area \,/\, nearest\_area$
3. $\varepsilon = 1 - intercepted\_proportion$
4. With probability ε:
5. return random action
6. If $nearest\_area == anxiety\_area$:
7. return $argmax(Q_{performance})$
8. If $nearest\_area == boredom\_area$:
9. return $argmin(Q_{performance})$

Output: action

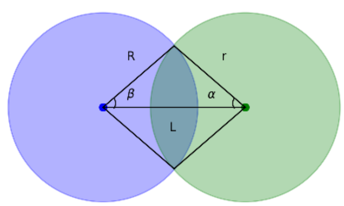

Fig. 5. Interception of boredom (purple) and current (green) areas

4. Experimental protocol

The experimental goal is to quantitively compare our approach with other approaches. We select running simulations because simulated users are appropriate to evaluate RL algorithms under defined human constraints (Bignold et al., 2021), which is our case. In addition, simulations allow objective comparisons and repeatability of experiments. Then, firstly, we set the problem by grading the skills of a real dataset publicly available to choose 5 of them from different levels of difficulty (see subsection 4.1). Second, we define data-driven users based on real data (see subsection 4.2). Third, we select the performance metrics (see subsection 4.3). Finally, we simulate 1000 runs and present mean values as results (see section 5).

4.1. Setting the problem

We selected the Cognitive Tutor 2006-2007 Bridge to Algebra dataset (Stamper & Pardos, 2016) because it is composed of enough records for our study (16.858 records from 587 students of 13-14 years old, during the 8th grade school year, performing 12 skills). Based on Minn et al. (2022), we apply Equation 6 to estimate the difficulty D for each skill belonging to the dataset by mapping the initial average success rate of a skill into 5 levels (from 1 to 5). In Equation 6, n represents the number of students who attempted the skill s_i, and O_j (1 if successful, 0 otherwise) is the outcome of the first attempt from student j to skill s_i. After applying the equation, to get the 5 skills graded that are required by the problem formulation (see paragraph 1 in section 3), we select one skill for each difficulty level (see Table 1): Plot whole number (difficulty 1), Calculate part in proportion with fractions (difficulty 2), Calculate unit rate (difficulty 3), Finding the intersection mixed (difficulty 4), and Plot imperfect radical (difficulty 5).

$$D(s_i) = (n+1) - round\left(\frac{\sum_{j=0}^{n} O_j}{n} \times 5\right) \tag{6}$$

4.2. Data-driven users

Our approach requires a user capable of solving tasks in which they must apply one of five skills and express emotions while doing so. For each task given, the data-driven user will provide a binary answer (1 if the solution is correct, 0 otherwise) and a dimensional emotion (valence and arousal to represent the mean emotion). For generating the answers, we select the Bayesian Knowledge Tracing model (see subsection 4.2.1), and for generating the emotions we use the AFEW-VA dataset (Kossaifi et al., 2017), composed of 600 videos displaying various facial expressions annotated per frame with levels of valence and arousal intensities in the range of -10 to 10 (see subsection 4.2.2). Finally, we integrate both models based on the user knowledge (see subsection 4.2.3).

4.2.1. Knowledge Tracing model

The Bayesian Knowledge Tracing model (Corbett & Anderson, 1994) is based on a Hidden Markov Model where the observable states are students' binary responses, and the hidden states are students' latent knowledge at a particular time step t. We apply expectation maximization (Pardos & Heffernan, 2010) to the Cognitive Tutor 2006-2007 Bridge to Algebra dataset to fit its four parameters prior, learn, guess, and slip, which are $P(L_0)$, $P(T)$, $P(G)$, and $P(S)$, respectively. From Yudelson et al. (2013), Equation 7 calculates the probability that a student correctly applies a skill, where $P(L_{t+1})$ is the probability of knowing the skill (also called the probability of skill mastery or skill proficiency) calculated in Equation 8, and $P(L_t|obs_t)$ is gotten in Equation 9 when the obs is correct or Equation 10 when incorrect.

$$P(C_{t+1}) = P(L_{t+1})\big(1 - P(S)\big) + (1 - P(L_{t+1}))P(G) \tag{7}$$

$$P(L_{t+1}) = P(L_t|obs_t) + (1 - P(L_t|obs_t))P(T) \tag{8}$$

$$P(L_t|obs_t = 1) = \frac{P(L_t)\big(1 - P(S)\big)}{P(L_t)\big(1 - P(S)\big) + (1 - P(L_t))P(G)} \tag{9}$$

$$P(L_t|obs_t = 0) = \frac{P(L_t)P(S)}{P(L_t)P(S) + (1 - P(L_t))(1 - P(G))} \tag{10}$$

Table 1 shows the initial probability of the answer being correct according to the difficulty for the skills selected with different difficulty levels (see subsection 4.1). Initial probabilities are consistent with the difficulty level, being the lower and the higher difficulties who have higher and lower probabilities of answering correctly as expected, respectively.

Table 1. Initial probabilities of answering correctly for each skill

Skill	**Difficulty**	$\boldsymbol{P(C_0)}$
Plot whole number	1	0.89
Calculate part in proportion with fractions	2	0.66
Calculate unit rate	3	0.50
Finding the intersection, Mixed	4	0.48
Plot imperfect radical	5	0.26

4.2.2. Emotional model

To generate emotions according to the flow theory, we define anxiety, boredom, and excitement areas according to the circumplex model of emotions (141, 242, and 48.6 degrees for anxiety, boredom, and excitement, respectively). In addition, we define two areas of transition between those emotional areas: anxiety to excitement, and excitement to boredom. Fig. 6 shows the mean emotions of 600 videos from the AFEW-VA dataset with the defined areas highlighted. Anxiety, boredom, excitement, anxiety to excitement, and excitement to boredom are composed of 59, 18, 68, 196, and 125 samples, respectively. The rest 134 samples were discarded (see yellow points in Fig. 6). For simulations, we chose a random point belonging to the wanted cluster. For example, if we want to simulate an expression of anxiety, then we randomly (uniform distribution) select one from the 59 samples that belong to anxiety.

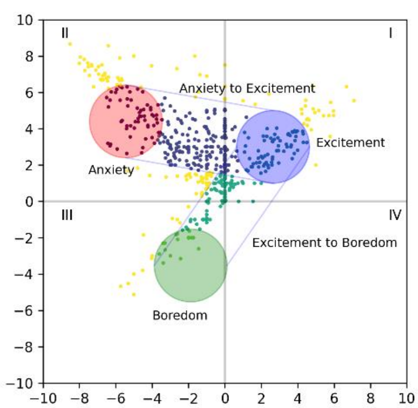

Fig. 6. Emotions in Anxiety and Boredom Areas.

4.2.3. Joining Knowledge Tracing and Emotions

A user that behaves according to the flow theory must follow three considerations. First, the user must express anxiety when his/her skill is low, and the challenge is high. Second, he/she must express boredom when his/her skill is high, but the challenge is low. Third, he/she expresses excitement about being in the flow. To know the user's skill, we can consider the probability of answering correct P(C) given by the BKT model, where a low probability means low skill and a high probability means high skill.

In addition, we can associate the challenge with the difficulty of the skill (see Table 1). It means that a low difficulty is a low challenge as well. Then, for connecting both knowledge and emotion models, we define Equation 11, where emotions are related to initial probabilities (gotten from fitting BKT models) of answering correct $P(C_0)$ for each difficulty level. For example, if the probability is lower or equal to 0.26 (initial probability of answering correct the skill *Plot whole number* considered as difficulty 1, as shown in Table 1), then the emotion will be anxiety.

$$emotion(p) = \begin{cases} anxiety & 0 < p \leq 0.26 \\ anxiety\ to\ excitemet & 0.26 < p \leq 0.48 \\ excitement & 0.48 < p \leq 0.66 \\ excitement\ to\ boredom & 0.66 < p < 0.89 \\ boredom & 0.89 < p \leq 1 \end{cases} \quad (11)$$

4.3. Performance metric

According to our target of measuring impact on users, we select learning gain as the main metric. For comparison with the other approaches where the target was measuring the agent performance, we select the utility because it is the metric that they use, and, to understand better the differences, we choose the skill selection rate. Finally, to show that our approach follows the three design considerations, we calculate the percentage of emotions by skill.

4.3.1. Learning gain. It is a useful metric for knowing how an intervention affects the student's knowledge (Hutchins et al., 2020). It requires two equivalent tests for comparing their scores: a pretest applied before the intervention and a posttest after it. Equation 12 shows how to calculate the learning gain, which is a real value in the range [0, 1], where max score is the total number of tasks in a test, pretest score is the number of tasks correctly solved in a test applied at the beginning, and posttest score is the number of tasks correctly solved in a test applied at the end. For our case, each test is composed of 15 tasks (3 tasks for each skill), being the max score equal to 15.

$$learning\ gain = \frac{posttest\ score - pretest\ score}{\max score - pretest\ score} \quad (12)$$

4.3.2. Utility. It is the metric used by Azoulay et al. (2014) for comparing the learning performance of agents. Although we do not pursue to maximize this utility, it will be useful to compare our approach with the others because their results are strongly dependent on this metric. Equation 13 shows that utility is the summation of difficulty D_i in which the tasks Q_i was correctly solved for all solved tasks n.

$$utility = \sum_{i=1}^{n} D_i |\; Q_i \; was \; correctly \; solved \tag{13}$$

4.3.3. Skill selection rate. It is useful for understanding better how the agents behave because it allows knowing tendencies for each one such as what skill favors each approach, allowing us to explain why each approach got its values of learning gain and utility. In addition, this metric is useful to identify unwanted behaviors. For example, we do not want a tendency to the easiest skill, which means a high probability of success but low values of learning gain. For each skill s, the selection rate is calculated with Equation 14, where the summation of the times that the skill s was selected is divided by the total of tasks n and multiplied by 100%.

$$selection\ rate_s = \frac{\sum_{i=1}^{n} 1 \mid s \; was \; selected}{n} \times 100\% \tag{14}$$

4.3.4. Percentage of emotions by skill. This metric allows us to confirm our three design considerations because we can get what skills tend to be selected when emotions are positives (consideration 1), anxiety or boredom (consideration 2), and close to anxiety or boredom (consideration 3). This percentage is calculated by dividing the emotional space into 400 hexagonal bins. Then, for each skill it is counted the number of emotions that are included in each bin b, divided by the total number of emotions n that precede the skill, and multiplied by 100% (see Equation 15). With this metric, we expect that medium difficulties tend to be selected when emotions are positives (consideration 1), and simplest or hardest difficulties be selected when anxiety or boredom (consideration 2) or when anticipating them (consideration 3).

$$percentage_b = \frac{\sum_{i=1}^{n} 1 \mid i \; \in b}{n} \times 100\% \tag{15}$$

5. Results

The results are presented according to each of the performance metrics defined in section 4.3. Each of them allows us to analyze a different aspect, such as the impact on users, or the performance of the agent.

5.1. Learning gain

From the user side, Fig. 7 presents the comparison of approaches according to the learning gain for 200 tasks. In general, our approach achieves better values of learning gain than those achieved by the other approaches. In the beginning (10 tasks), our approach achieves 0.09 of learning gain, followed by DVRL with 0.07, and Q-learning and Virtual Learning with 0.065. The Bayesian approach achieves a lower value of around 0.03. Later, until the end, our approach keeps increasing the learning gain until around 0.49 at task 200. Considering that the maximum learning gain is 1 (it happens when the pretest is 0 and the posttest is equal to the max score, meaning that a student does not know at all in the pretest and learns all in the posttest), a value of 0.49 means significant progress. Virtual Learning, DVRL, and Q-learning also increase until the end but achieve lower values (0.32, 0.29, and 0.28, respectively). In contrast, the Bayesian approach keeps around 0.1 rather than increasing, achieving the worst results. Then, using our approach allows more benefits to students than the other approaches. In addition, Fig. 7 shows how many tasks a student must solve to achieve a wanted learning gain. For example, if a student wants to gain around 0.3, he must solve around 50 and 60 tasks.

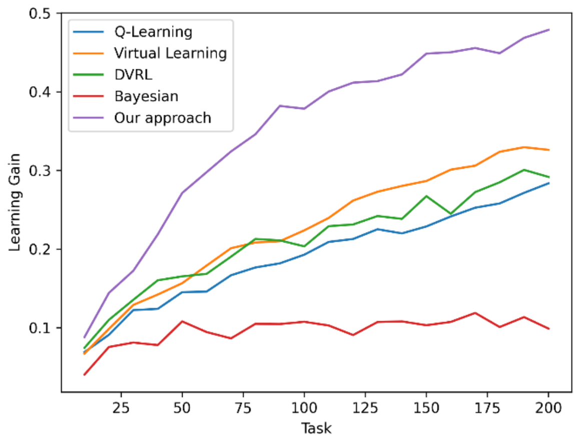

Fig. 7. Comparison of approaches according to the learning gain

5.2. Utility

From the agent side, Fig. 8 shows the comparison of approaches according to the utility metric, where the Bayesian approach achieves better values than the others. Although it is not a fair comparison for our approach because our algorithm does not pursue to optimize the utility, this metric is useful because it can give us information about the performance of our approach in the context of the related works, having as a reference that the maximum possible value is 1000 (it happens when all the tasks belong to the maximum difficulty). The other approaches focus on maximizing this metric because they want an agent that learns the skill with a high difficulty level in which users frequently respond correctly, assuming that it is the best case for the students' learning. We find that our algorithm behaves better than Virtual Learning and is close to the other approaches (520, 500, 490, and 450 for Bayesian, DVRL, Q-learning, and our approach, respectively). It is interesting because our algorithm pursues to adjust difficulty based on student emotions rather than utility. It means that emotions as feedback guide indirectly in the same direction that the utility as a guide, being possible because a user that behaves according to the flow theory tends to pursue positive emotions that are related to medium difficulties (around difficulty 3) which, in this case, is close to the higher difficulty level in which users frequently respond correctly (difficulty 4 according to the Bayesian approach).

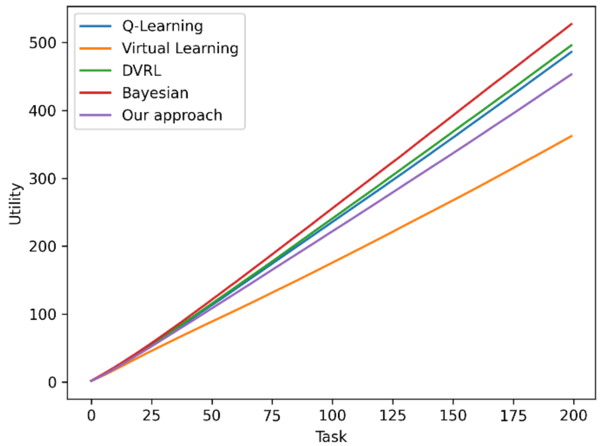

Fig. 8. Comparison of approaches according to the utility

5.3. Skill selection rate

To understand better the differences between approaches, we analyze the distribution of skill selection (see Fig. 9). We find that our approach tends to select skills of intermediate difficulty (25%, 40%, and 30%, for difficulties 2, 3 and 4, respectively), avoiding the easiest and hardest but trying those a little bit (around 2% and 7% for difficulties 1 and 5, respectively). Those results are consistent with the flow theory because it is wanted to avoid boredom and anxiety, which are related to the easiest and hardest difficulties. In contrast, the Bayesian approach tends to select skill 4 (almost 100% of the time), which is consistent with its target of finding one skill that maximizes the utility, as shown in Fig. 8. On the other hand, the Virtual Learning approach tends to select skill 2 around 50% of the time and favors skills 1 and 3 in comparison with skills 4 and 5 (see Fig. 9). Finally, Q-learning has a similar tendency to DVRL because it moderately tries all the skills and favors the skill 4 as well, but as shown in Fig. 9, Q-learning tries a little bit less skills 4 and 5, getting lower utility values (see Fig. 8.). That result was expected because DVRL is an enhanced version of Q-learning where the updating phase of the Q values is applied to the neighboring difficulties levels as well.

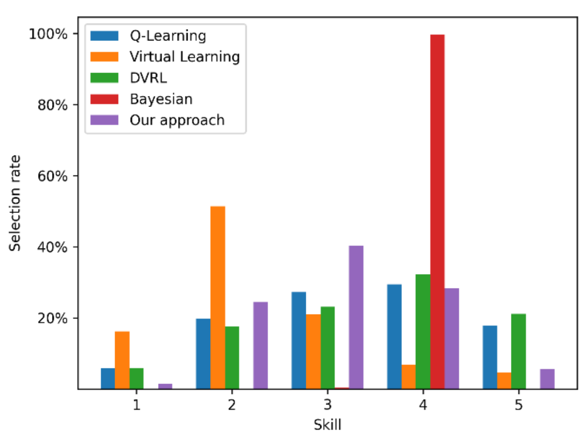

Fig. 9. Comparison of approaches according to the skill selection rate

5.4. Percentage of emotions by skill

Fig. 10 presents the percentage of emotions that precede each skill to show how our approach follows the three design considerations. First, about keeping the flow channel that is related to positive emotions, skill 3 (which is the skill with higher selection rate in our approach according to Fig. 9) tends to be selected from positive emotions, as shown in the first quadrant of plot c in Fig.10. In contrast, skill 1 is selected very few from positives emotions, and skills 2, 4, and 5 moderately (in Fig. 10, see plots a, b, d, and e, respectively). Because the skill with a higher selection rate is the most selected from positive emotions, we argue that our approach tends to keep the flow channel (that is consistent with a review of the literature (Peifer et al., 2022) that concludes that, in general, studies show that flow channel is associated with higher positive emotional states). Second, about managing anxiety or boredom by looking at the easiest or hardest skills, we see in the second quadrant of plot a of Fig. 10 that effectively the most emotions of anxiety precede skill 1, which is consistent with our algorithm that looks for the easiest skill when anxiety. Similarly, we see in the third quadrant of plot e that the most emotions that precede skill 5 are around boredom indicating that it is looking for the higher difficulty when the user is bored. Third, about anticipating anxiety or boredom, which is related to emotions in regions A2 and B2, we see in all plots of Fig. 10 that A2 is a low-frequented area for emotions, which can be explained by the approach of our algorithm that pursues increase learning gains, then, it is expected that B2 be the most frequented area because a user who tends to learn also tends to be bored. For that reason, our results show more activity in B2, which means that our approach tends to keep the users learning but avoids boredom.

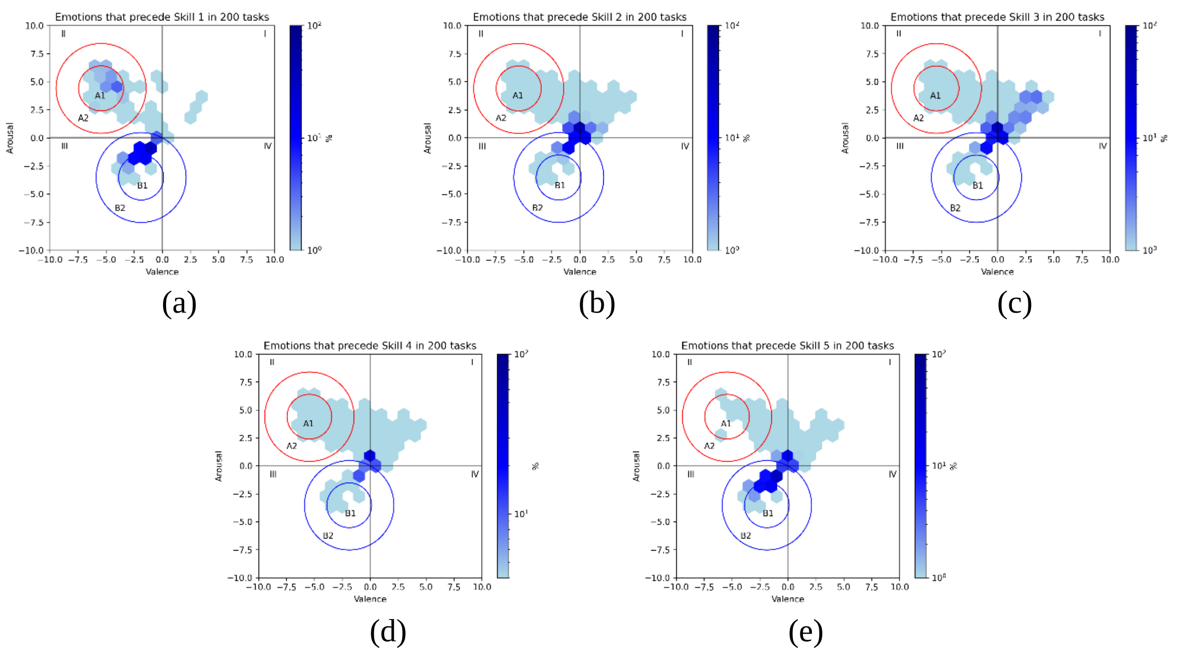

Fig. 10. Skill selection according to the emotions

6. Comparison with related works

We found six related works (Chan & Nejat, 2012; D’Mello et al., 2010; D’Mello et al., 2012; Gordon et al., 2016; Park et al., 2019; Pérez et al., 2023) that consider student emotions to determine the pedagogical policy as our approach does. All of them have different objectives, but all are associated with Intelligent Tutoring Systems (ITSs). Naturally, objectives are different because ITSs focus on different strategies. For example, they focus on memory games (Chan & Nejat, 2012), dialogues (D’Mello et al., 2010), storytelling (Park et al., 2019), and problem-solving (Pérez et al., 2023). Our approach focuses on problem-solving but rather than adapting the topic like Pérez et al. (2023), we adapt the skill that is related to the problem's difficulty. Those differences do not allow for comparing those approaches numerically. We compare five characteristics related to emotions: emotion representation, emotion incorporation, pursuing positive emotions, managing anxiety or boredom, and anticipating anxiety or boredom (see Table 2). Emotion representation is related to how emotions are numerically represented; emotion incorporation is how the emotional value is considered in the algorithm to make the decision. The rest of the characteristics verify whether the approaches follow our three design considerations: pursue positive emotions, manage anxiety and boredom, and anticipate anxiety or boredom.

Table 2. Comparison of related works.

Approach	Emotion representation	Emotion incorporation	Pursue positive emotions	Manage anxiety or boredom	Anticipate anxiety or boredom
Q-learning (Park et al., 2019)	A discrete model of four categories: q1, q2, q3, and q4.	State	No	No	No
SARSA (Gordon et al., 2016)	A discrete model of three categories: neg, med, and pos.	State and reward	Yes	No	No
MAXQ (Chan & Nejat, 2012)	A discrete model of four categories: stressed, neutral, excited, and pleased.	State	No	No	No
Q-learning (Pérez at al., 2023)	Dimensional model of emotional valence.	Reward shaping	Yes	No	No
Hand-coded rules (D’Mello et al., 2010)	boredom, confusion, frustration, and neutral	Rules	Yes	Yes	No
Hand-coded rules (D’Mello et al., 2012)	Dimensional model of two dimensions: pleasure-displeasure and arousal-sleepiness.	Rules	Yes	Yes	No
Our approach based on Q-learning	Dimensional model of valence and arousal	Exploration-exploitation strategy and reward	Yes	Yes	Yes

About the approaches, we argue that hand-coded methods (D’Mello et al., 2010; D’Mello et al., 2012) are limited because students’ knowledge is different and changes over time. In contrast, the rest of the approaches, which are based on RL, fit better for adjusting parameters because it learns automatically from experiences. We note that emotion representation is related to emotion incorporation. Particularly, for tabular algorithms like SARSA (Gordon et al., 2016), Q-learning (Park et al., 2019; Pérez et al., 2023), and MAXQ (Chan & Nejat, 2012), incorporating emotions as states is limited to discrete emotions; however, rather than using discrete emotions, the methods use a continuous space to represent emotions, being mandatory to convert the continuous value to discrete categories like quartiles (Park et al., 2019) or three categories (Gordon et al., 2016). On the other hand, incorporating emotions as a reward shaping function allows using a continuous space, like is used by Pérez at al. (2023). Nonetheless, our approach proposes incorporating emotional values in the exploration-exploitation strategy for managing and anticipating boredom and anxiety. So, according to the circumplex model, we use the circular area around the point created from the valence and arousal values for detecting anxiety and boredom.

About our three design considerations, several related works pursue positive emotions (D'Mello et al., 2010; D'Mello et al., 2012; Gordon et al., 2016; Pérez et al., 2023). Approaches based on hand-coded rules (D'Mello et al., 2010;D'Mello et al., 2012) do it indirectly because they focus on detecting some negative emotions to be managed to get positive emotions. In contrast, approaches based on RL (Gordon et al., 2016; Pérez et al., 2023) pursue positive emotions directly because they try to optimize the task by maximizing positive emotional states. Specifically, Gordon et al. (2016) incorporate the emotional valence in the reward and Pérez et al. (2023) uses both valence and arousal to feed a reward-shaping function. Like Gordon et al. (2016) do, our approach incorporates emotion valence as a reward to be maximized but, it also incorporates both valence and arousal in the exploration-exploitation strategy to manage negative unwanted emotions as well. That is, our approach pursues positive emotions directly and indirectly.

Only the related works with hand-coded approaches manage unwanted emotions. Specifically, D'Mello et al. (2010) detect learners' boredom, confusion, and frustration, and then, it applies rules for managing those. For example, if the current state is classified as boredom and the previous state was classified as frustration, it shows a random message to the user like this: "Maybe this topic is getting old. I'll help you finish so we can try something new". On the other hand, D'Mello et al. (2012) identify when the student is bored, disengaged, or zoning out, to attempt to reengage the student with dialog moves that direct the student to reorient his attentional patterns. For example, if it identifies boredom, then the message could be ''Please pay attention'' or ''I'm over here you know''. Our approach manages unwanted emotions by adjusting the difficulty of the task, that is, if it detects boredom the task will be the hardest but if it identifies anxiety the task will be the easiest because we try to get out the student from unwanted emotions as fast as possible.

None of the related works anticipate unwanted emotions. Our approach does anticipate those by detecting emotions that are close to the unwanted emotions and later manages those with a strategy based on $\varepsilon - greedy$ to assure exploration. It means that if we detect an emotion close to an unwanted emotion, then we manage it either by selecting the hardest or easiest task (according to the case) or choosing a random difficulty. We claim the other approaches that manage unwanted emotions (D'Mello et al., 2010; D'Mello et al., 2012) could anticipate emotions as well by detecting close emotions to the unwanted emotions.

7. Conclusions

An approach based on flow theory for considering emotions as implicit feedback in the RL framework was proposed to adapt difficulty in the context of tutoring systems. Our proposal includes the students' emotions in the exploration-exploitation strategy and the reward of the RL framework. On the one hand, it uses both valence and arousal for localizing a point in the space according to the circumplex model of emotions, but creating a circle so that points into the area are considered as a group. Then, if the student's emotion belongs to the anxiety area, then anxiety is detected to be managed, but if its area intercepts the anxiety area, then anxiety is anticipated. On the other hand, it uses valence as a reward to learn what action maximizes positive values.

Results show that our proposal covers the three design considerations. First, it pursues positive emotions because the skill with the highest selection rate (see skill 3 in Fig. 9) is the most selected from positive emotions (see Fig. 10). It knows what skills are related to positive emotions because of the valence as a reward that we apply to learn Q_{flow}. Second, it manages anxiety and boredom as shown in Fig. 10 because emotions in the anxiety area are followed by the easiest skill, and emotions in the boredom area are followed by the hardest skill. It is possible because our approach gets from $Q_{performance}$ what is the easiest through $argmax(Q_{performance})$ and the hardest through $argmin(Q_{performance})$. Third, it anticipates anxiety and boredom as presented in Fig. 10 where emotions in the B2 area are the more frequent avoiding boredom. Anticipation is possible because we detect if the current emotion area intercepts the unwanted area to consider applying the easiest or hardest skill according to the case.

Results also show that our approach is better than the others in fostering student learning gains in the context of adapting difficulty for tutoring systems. Other approaches tend to foster learning gains slowly as Virtual Learning while others converge to low values as the Bayesian (see Fig. 7). The reason is that our approach tends to select intermediate difficulty (skill 3), Virtual Learning an easier (skill 2), and Bayesian the penultimate level of difficulty (see skill 4 in Fig. 9). That is, Bayesian focuses on almost the more difficult skill that is expected to have lower probability of success than the majority but receives a high reward when success. Virtual Learning focuses on skill 2 that is the second level of difficulty, so the probability of success is expected to be higher than the majority. In contrast, our approach focuses on skill 3 that is expected to be a balance between the probability of success and the value of the reward in comparison with the other skills. In other words, enhancing mastery in intermediate levels allows getting higher learning gains than focusing on higher or lower levels.

A limitation was found in this research because there is no available dataset where students' dimensional emotions as in the circumplex model are collected while students solve different tasks. We addressed that issue by joining two datasets, however, according to the task, students could have different emotional behaviors, being necessary to study other contexts. In addition, we identify two useful future works. On the one hand, we studied how to incorporate emotions in the RL framework to foster student learning gain, but a similar study is needed to know how to incorporate emotions in the context of transfer learning, which implies that students can transfer their acquired skills to new situations as problems that require several skills at the same time or problems of new skills. On the other hand, considering that adapting feedback that students receive during training has shown promise (Zahabi & Abdul, 2020), we could include in our approach feedback support and manage it by adapting its parameters (timing, content, and modality) to enhance even more the student learning gains.

Data availability

The Cognitive Tutor 2006-2007 Bridge to Algebra dataset is available at https://pslcdatashop.web.cmu.edu/KDDCup/downloads.jsp. The AFEW-VA dataset is available at https://ibug.doc.ic.ac.uk/resources/afew-va-database/.

8. References

Akalin, N., & Loutfi, A. (2021). Reinforcement Learning Approaches in Social Robotics. *Sensors, 21*(4), 1292. https://doi.org/10.3390/s21041292

Aleven, V., McLaughlin, E. A., Glenn, R. A., & Koedinger, K. R. (2017). Instruction based on adaptive learning technologies. In R. E. Mayer & P. Alexander, *Handbook of research on learning and instruction*, New York. https://doi.org/10.4324/9781315736419

Ausin, M. S. (2019). Leveraging deep reinforcement learning for pedagogical policy induction in an intelligent tutoring system. In *Proceedings of the 12th International Conference on Educational Data Mining (EDM 2019)*, Montreal, Canada. https://par.nsf.gov/biblio/10136494

Azoulay, R., David, E., Hutzler, D. & Avigal, M. (2014). Adaptation Schemes for Question's Level to be Proposed by Intelligent Tutoring Systems. In *International Conference on Agents and Artificial Intelligence*, Angers, France. https://doi.org/10.5220/0004732402450255

Bignold, A., Cruz, F., Dazeley, R., Vamplew, P., & Foale, C. (2021). An Evaluation Methodology for Interactive Reinforcement Learning with Simulated Users. *Biomimetics, 6*(1), 13. https://doi.org/10.3390/biomimetics6010013

Chan, J., & Nejat, G. (2012). Social Intelligence for a Robot Engaging People in Cognitive Training Activities. *International Journal of Advanced Robotic Systems*, *9*(4). https://doi.org/10.5772/51171

Chi M., VanLehn K., Litman D., & Jordan P. (2010). Inducing Effective Pedagogical Strategies Using Learning Context Features. In De Bra P., Kobsa A., Chin D., *User Modeling, Adaptation, and Personalization*. Lecture Notes in Computer Science, Vol. 6075. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-13470-8_15

Corbett, A.T. & Anderson, J.R. (1994). Knowledge tracing: Modeling the acquisition of procedural knowledge. User modeling and user-adapted interaction, 4, 253-278. https://doi.org/10.1007/BF01099821

Cruz, F., Magg, S., Weber, C., & Wermter, S. (2016). Training agents with interactive reinforcement learning and contextual affordances. *IEEE Transactions on Cognitive and Developmental Systems*, *8*(4), 271-284. https://doi.org/10.1109/TCDS.2016.2543839

Csikszentmihalyi, M. (1990). Flow: The Psychology of Optimal Experience. *Journal of Leisure Research*, *24*(1), 93–94. https://doi.org/10.2307/258925

Cuartas, C., Aguilar, J. (2023) Hybrid algorithm based on reinforcement learning for smart inventory management. *J Intell Manuf* 34, 123–149. https://doi.org/10.1007/s10845-022-01982-5

D'Mello, S., Lehman, B., Sullins, J., Daigle, R., Combs, R., Vogt, K. & Graesser, A. (2010). A time for emoting: When affect-sensitivity is and isn't effective at promoting deep learning. In

Intelligent Tutoring Systems: 10th International Conference, ITS 2010, Pittsburgh, USA. https://doi.org/10.1007/978-3-642-13388-6_29

D'Mello, S., Olney, A., Williams, C., & Hays, P. (2012). Gaze tutor: A gaze-reactive intelligent tutoring system. *International Journal of human-computer studies, 70*(5), 377-398. https://doi.org/10.1016/j.ijhcs.2012.01.004

Ekman, P., & Friesen, W. V. (1971). Constants across cultures in the face and emotion. *Journal of Personality and Social Psychology, 17*(2), 124–129. https://doi.org/10.1037/h0030377

Fraulini, N.W., Marraffino, M.D., Garibaldi, A.E. (2023). Identifying Individual Differences that Predict Usage of an Adaptive Training System in a United States Marine Corps Course. In Sottilare, R.A., Schwarz, J., *Adaptive Instructional Systems. HCII 2023*. Lecture Notes in Computer Science, Cham. https://doi.org/10.1007/978-3-031-34735-1_16

Gordon, G., Spaulding, S., Kory Westlund, J., Lee, J. J., Plummer, L., Martinez, M., Das, M., & Breazeal, C. (2016). Affective Personalization of a Social Robot Tutor for Children's Second Language Skills. *Proceedings of the AAAI Conference on Artificial Intelligence, 30*(1). https://doi.org/10.1609/aaai.v30i1.9914

Hutchins, N.M., Biswas, G., Maróti, M., Lédeczi, Á., Grover, S., Wolf, R., Blair, K.P., Chin, D., Conlin, L., Basu, S. & McElhaney, K. (2020). C2STEM: A system for synergistic learning of physics and computational thinking. *Journal of Science Education and Technology, 29*, 83-100. https://doi.org/10.1007/s10956-019-09804-9

Johri, A. (2023). *International Handbook of Engineering Education Research* [1st ed.]. Routledge. https://doi.org/10.4324/9781003287483

Kang, H., Sales, A. & Whittaker, T. (2024). Flow with an intelligent tutor: A latent variable modeling approach to tracking flow during artificial tutoring. *Behavior Research Methods, 56*, 615-638. https://doi.org/10.3758/s13428-022-02041-w

Kossaifi, J., Tzimiropoulos, G., Todorovic, S. & Pantic, M. (2017). AFEW-VA database for valence and arousal estimation in-the-wild. *Image and Vision Computing, 65*, 23-36. https://doi.org/10.1016/j.imavis.2017.02.001

Landowska, A. (2018). Towards New Mappings between Emotion Representation Models. *Applied Sciences, 8*(2), 274. https://doi:10.3390/app8020274

McDuff, D., Mahmoud, A., Mavadati, M., Amr, M., Turcot, J., & Kaliouby, R. E. (2016). AFFDEX SDK: a cross-platform real-time multi-face expression recognition toolkit. In *Proceedings of the 2016 CHI conference extended abstracts on human factors in computing systems*, California, USA. https://doi.org/10.1145/2851581.2890247

Minn, S., Vie, J.-J., Takeuchi, K., Kashima, H., & Zhu, F. (2022). Interpretable Knowledge Tracing: Simple and Efficient Student Modeling with Causal Relations. *Proceedings of the AAAI*

Conference on Artificial Intelligence, 36(11), 12810-12818. https://doi.org/10.1609/aaai.v36i11.21560

Ortony, A., Clore, G. L., & Collins, A. (1988). *The cognitive structure of emotions*. Cambridge University Press. https://doi.org/10.1017/CBO9780511571299

Pardos, Z., & Heffernan, N. (2010). Navigating the parameter space of Bayesian Knowledge Tracing models: Visualizations of the convergence of the Expectation Maximization algorithm. *In Proceedings of the 3rd International Conference on Educational Data Mining*, Pittsburgh, Pennsylvania.

Park, H. W., Grover, I., Spaulding, S., Gomez, L., & Breazeal, C. (2019). A Model-Free Affective Reinforcement Learning Approach to Personalization of an Autonomous Social Robot Companion for Early Literacy Education. *Proceedings of the AAAI Conference on Artificial Intelligence, 33*(01), 687-694. https://doi.org/10.1609/aaai.v33i01.3301687

Peifer C., Wolters G., Harmat L., Heutte J., Tan J., Freire T., Tavares D., Fonte C., Andersen F., van den Hout J., Šimleša M., Pola L., Ceja L. & Triberti S. (2022). A Scoping Review of Flow Research. *Frontiers in Psychology, 13*. https://doi.org/10.3389/fpsyg.2022.815665

Pérez, J., Aguilar, J. & Dapena, E. (2018). MIHH: Un modelo de interacción humano- humano. *Revista Venezolana de Computación, 5*(1), 10-19.

Pérez, J., & Castro, J. (2018). LRS1: un robot social de bajo costo para la asignatura “Programación 1”. *Revista colombiana de tecnologías de avanzada (RCTA), 2*(32), 68-77. https://doi.org/10.24054/16927257.v32.n32.2018.3028

Pérez, J., Aguilar, J. & Dapena, E. (2020). MIHR: A Human-Robot Interaction Model. *IEEE Latin America Transactions, 18*(9), 1521-1529. https://doi.org/10.1109/TLA.2020.9381793

Pérez J., Dapena E., Aguilar J. & Carrillo G. (2022). Reinforcement Learning for Estimating Student Proficiency in Math Word Problems. In *2022 XVII Latin American Conference on Learning Technologies (LACLO)*, Armenia, Colombia. http://doi.org/10.1109/LACLO56648.2022.10013399

Pérez, J., Aguilar, J. and Dapena, E. (2023). Affective Observation based on Reinforcement Learning for an Adaptive Tutoring System for Math Word Problems. [Manuscript submitted for publication]. Department of Computing, University at Los Andes.

Ritter, F. E., Nerb, J., Lehtinen, E., & O’Shea, T. M. (2007). *In order to learn: How the sequence of topics influences learning*. Oxford University Press.

Russell, J. A. (1980). A circumplex model of affect. Journal of Personality and Social *Psychology, 39*(6), 1161–1178. https://doi.org/10.1037/h0077714

Salazar C., Aguilar J., Monsalve-Pulido J., Montoya E. (2021) Affective recommender systems in the educational field. A systematic literature review, *Computer Science Review*, 40, https://doi.org/10.1016/j.cosrev.2021.100377.

Sepulveda, G.K., Besoain, F. & Barriga, N.A. (2019). Exploring dynamic difficulty adjustment in videogames. In *2019 IEEE CHILEAN Conference on Electrical, Electronics Engineering, Information and Communication Technologies (CHILECON)*, Valparaiso, Chile. https://doi.org/10.1109/CHILECON47746.2019.8988068

Seyderhelm, A. & Blackmore, K. (2021). Systematic Review of Dynamic Difficulty Adaption for Serious Games: The Importance of Diverse Approaches. Retrieved January 20, 2024, from http://dx.doi.org/10.2139/ssrn.3982971

Shen, S., Mostafavi, B., Barnes, T., & Chi, M. (2018). Exploring Induced Pedagogical Strategies Through a Markov Decision Process Framework: Lessons Learned. *Journal of Educational Data Mining*, *10*(3), 27-68. https://doi.org/10.5281/zenodo.3554713

Shernoff, D. J., Csikszentmihalyi, M., Shneider, B., & Shernoff, E. S. (2003). Student engagement in high school classrooms from the perspective of flow theory. *School Psychology Quarterly*, *18*(2), 158–176. https://doi.org/10.1521/scpq.18.2.158.21860

Stamper, J., & Pardos, Z. A. (2016). The 2010 KDD Cup Competition Dataset: Engaging the machine learning community in predictive learning analytics. *Journal of Learning Analytics*, *3*(2), 312-316. https://doi.org/10.18608/jla.2016.32.16

Sutton, R. S., & Barto, A. G. (2018). *Reinforcement learning: An introduction*. MIT Press.

Tsiakas, K., Abujelala, M., & Makedon, F. (2018). Task Engagement as Personalization Feedback for Socially-Assistive Robots and Cognitive Training. *Technologies*, *6*(2), 49. https://doi.org/10.3390/technologies6020049

Van Otterlo, M., Wiering, M. (2012). Reinforcement Learning and Markov Decision Processes. In Wiering, M., van Otterlo, M. (eds) *Reinforcement Learning. Adaptation, Learning, and Optimization*, Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-27645-3_1

Vanlehn, K. (2006). The behavior of tutoring systems. *International Journal of Artificial Intelligence in Education*, *16*(3), 227–265.

Watkins, C. J., & Dayan, P. (1992). Q-learning. *Machine learning*, *8*(3), 279-292. https://doi.org/10.1007/BF00992698

Wang, Y., Song, W., Tao, W., Liotta, A., Yang, D., Li, X., Gao, S., Sun, Y., Ge, W., Zhang, W. & Zhang, W. (2022). A systematic review on affective computing: Emotion models, databases, and recent advances. *Information Fusion*, *83*, 19-52. https://doi.org/10.1016/j.inffus.2022.03.009

Yudelson, M.V., Koedinger, K.R. & Gordon, G.J. (2013). Individualized bayesian knowledge tracing models. In Lane, H.C., Yacef, K., Mostow, J., Pavlik, P. (eds) Artificial Intelligence in

Education. AIED 2013. *Lecture Notes in Computer Science*, Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-39112-5_18

Zahabi, M., Abdul Razak, A.M. (2020). Adaptive virtual reality-based training: a systematic literature review and framework. *Virtual Reality*, *24*, 725–752. https://doi.org/10.1007/s10055-020-00434-w

Zhou, G., Azizsoltani, H., Ausin, M.S., Barnes, T. & Chi, M. (2022). Leveraging granularity: Hierarchical reinforcement learning for pedagogical policy induction. *International journal of artificial intelligence in education*, *32*(2), 454-500. https://doi.org/10.1007/s40593-021-00269-9

Zini, F., Le Piane, F. & Gaspari, M. (2022). Adaptive Cognitive Training with Reinforcement Learning. *ACM Transactions on Interactive Intelligent Systems*, *12*(1), 1-29. https://doi.org/10.1145/3476777

Zohaib M. & Nakanishi, H. (2018). Dynamic Difficulty Adjustment (DDA) in computer games: A review. *Advances in Human-Computer Interaction*, *2018*. https://doi.org/10.1155/2018/5681652