A Reinforcement Learning-Based Adaptive Learning System

Doaa Shawky(✉) and Ashraf Badawi

Center for Learning Technologies, University of Science and Technology,
Zewail City, Giza, Egypt
{dshawky, abadawi}@zewailcity.edu.eg

Abstract. With the plethora of educational and e-learning systems and the great variation in students' personal and social factors that affect their learning behaviors and outcomes, it has become mandatory for all educational systems to adapt to the variability of these factors for each student. Since there is a large number of factors that need to be taken into consideration, the task is very challenging. In this paper, we present an approach that adapts to the most influential factors in a way that varies from one learner to another, and in different learning settings, including individual and collaborative learning. The approach utilizes reinforcement learning for building an intelligent environment that, not only provides a method for suggesting suitable learning materials, but also provides a methodology for accounting for the continuously-changing students' states and acceptance of technology. We evaluate our system through simulations. The obtained results are promising and show the feasibility of the proposed approach.

Keywords: Adaptive learning · Reinforcement Learning
Computer-supported collaborative learning

1 Introduction

Personalized learning often refers to the individualized instruction and support provided to students, which usually involves the integration of technology in a blended learning scenario [1]. The concept is viewed as the new approach to learning in which "one-size-fits- all" strategy is no longer applicable or acceptable [2]. Personalized learning encompasses several strategies. Usually, student's progress towards a clearly-defined goal is continuously monitored and assessed. In addition, students are provided with personalized learning paths, and they have frequently-updated profiles with weaknesses, strengths, motivation and goals.

In order to provide the aforementioned strategies, we need to build an intelligent learning environment that continuously monitors the variables that affect learning in different settings, and hence, update the suggested learning paths and materials from one learner to another. This is a non-trivial task, since the factors that affect learning can not be modeled or measured in isolation from each other [3–7]. In addition, they are mediated by other factors that may be hidden or unclear. For example, the learning

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 221–231, 2018.
https://doi.org/10.1007/978-3-319-74690-6_22

experience is influenced by learners' affective states, which might not be easily-measured or monitored.

The literature includes several studies that provide promising approaches to personalized learning. For instance in [8], ontology-based models for students, learning objects, and teaching method are proposed. The models consist of four layers that support personalized learning through reasoning and rule-based actions. Also in [9], a personalized learning process is supported by tuning the compatibility level of the learning objects with respect to the learning style of the learner. In addition, the complexity level of the learning objects with respect to the knowledge level of the learner and her interactivity level during the learning process using a modified form of genetic algorithm is modified. Results show the improvement in students' satisfaction. Moreover, in [10], case base planning techniques are used to generate sequences of e-learning routes which are tailored to the students' profiles. Also in [11], a survey on students' modeling approaches for building an automatic tutoring system is presented. The study concluded that for the different modeling tools and methods used, the most common-modeled student's characteristic is the knowledge level and the least common-modeled student's characteristic is her/his meta-cognitive features. However, detecting which set of characteristics is more important is still an open question.

This study develops a framework for personalized learning systems that alleviates some of the shortcomings and challenges to building an effective personalized learning system. The framework is based on the unsupervised machine learning tool; the reinforcement learning (RL). Since personalized learning systems have to be highly-dynamic, RL would be an effective tool for modeling the features of such systems. This is mainly because RL has the potential of dynamically approximating a changing model of the environment. The proposed approach consists of the following steps. Firstly, the learner's state is determined. Secondly, a learning material or path is suggested through a set of actions. Thirdly, based on reinforcement learning, the learner state is updated, in addition, the rewards received by recommended learning paths or material are updated.

The rest of the paper is organized as follows. In Sect. 2, a review on RL is provided. In Sect. 3, the proposed RL-based approach is presented. In addition, in Sect. 4, the system is evaluated and simulation experiments and results are discussed. Finally, Sect. 5 presents the conclusions and outline directions for future work.

2 Reinforcement Learning

Reinforcement learning is inspired by how learning occurs naturally by interacting with the environment, and by how biological systems learn [12]. Similar to all types of learning, it is about mapping situations to actions in order to maximize some rewards. However, the challenge in this type of learning is that, as opposed to other machine learning paradigms, the learner has to discover by herself the best action to be taken in a given situation. Thus, a learning agent must be able to sense the environment and choose the action that would maximize the rewarding function and update her state accordingly. In addition, she has to operate despite the uncertainty about the environment she might have.

As reinforcement learning schemes build environment information through exploration, they are suitable for unsupervised online implementation. A general RL is shown in Fig. 1. The environment can be characterized by the configuration or values of a certain number of its features, which is called its state, denoted at time t as S(t). Each state has a value, dependent upon a certain immediate reward or cost, denoted at time t as R(t), which is generated when it is entered. At each time instance, the agent may take one of a number of possible actions, A(t), which affects the next state of the system, S(t + 1), and therefore the next reward/cost experienced, according to certain transition probabilities. The agent's choice of actions, given the current state of the system, is modified by experience. Thus, an RL system uses its past experience of action taken in a certain system state and reward experienced to update its decision for future actions. A policy of actions to be taken given particular system states is developed over time by the agent as it interacts with the environment.

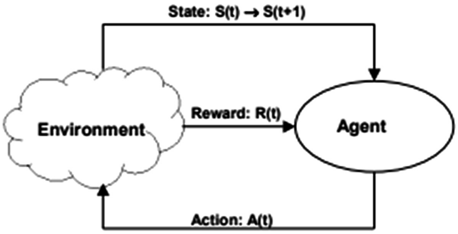

Fig. 1. An RL system [12]

The reinforcement learning problem is usually solved by dynamic programming, Monte Carlo methods, or temporal difference methods (TD) which is a combination of Monte Carlo and dynamic programming [13]. In TD learning, no model is used for mimicking the environment, however the learnt rewards are updated. The main objective is to estimate the value function V_π for a given policy π, which is called the prediction problem. Similar to Monte Carlo methods, TD uses experience to update the estimate of v for the states occurring in that experience. However, in Monte Carlo methods, the updates are done when the return following the visit is known. This is not the case in TD, where the method waits for the next time step $t + 1$ to update the observed reward R_{t+1} and the estimate $V(S_{t+1})$. The simplest TD method is given by (1):

$$V(S_t) \leftarrow V(S_t) + \alpha(R_{t+1} + \gamma V(S_{t+1}) - V(S_t)) \tag{1}$$

Another commonly used method for solving an RL problem is the Q-learning [14]. This algorithm allows learning the optimal policy to accomplish, based on the history of interactions of the system with the environment. In contrast with TD, this algorithm is an off-policy algorithm because no policy is used for suggesting the actions.

The actions are suggested based on some other criterion. This if the system is in state S_i, and it takes the action a_i, it will obtain a reward of r_{i+1}. Each time the system takes an action, given a state, and it receives a reward, an estimation of the scores the state S receives under the action a, denoted by Q(s, a), which is updated based on (2):

$$Q(s,a) \leftarrow Q(s,a) + \alpha(r\ +\ \gamma\ \mathrm{maxa}'Q(s', a') - \ Q(s,a)) \tag{2}$$

where α is a step rate; r is the observed reward, s’ is the new state, $\gamma < 1$ is a discounted factor for the future rewards received under the taken action. $Q(s', a')$ is the estimation of the maximum reward that system can measure by taking some future action in the state s'. The complete algorithm is given below.

```
Initialize Q(s, a) arbitrarily
Repeat
  Initialize s
  Repeat
    Choose a from s using $\epsilon$-greedy policy
    Take action a, measure r and observe $s'$
    $Q(s, a) \leftarrow Q(s, a) + \alpha(r + \gamma\ max_{a'}\ Q(s', a') - Q(s, a))$
    $s \leftarrow s'$
```

The Q-learning Algorithm

3 Proposed Approach

This section describes the framework of the proposed system and analyses its main components.

3.1 Main Components

The framework consists of six main components as shown in Fig. 2. The six components are connected to a user interface, and students’ database. The system starts by loading the student’s static information, in addition to the state-action matrix history. This includes some static data (e.g., gender, major, courses, etc.), in addition to some dynamic data (e.g., state-action-reward history, interactions level, log activities, etc.). Student state is loaded in Step 2. Initially, this represents her state the last time she was logged into the system. If this is her first time, a state that matches her static data is assigned to her. In the third step, an action is suggested, which usually includes a recommended learning material or some engaging material such as some pieces of advice from her instructor in a written, or recorded video or audio, a quotation, or even a joke. In the fourth step, the reward of the taken action is measured. This includes a direct reward where the student is asked to provide a value for her satisfaction level about the recommended material. Moreover, her interactivity level with the system is measured and combined with her satisfaction level to update the reward. Both of these actions are assigned a value out of 5, and these values are used to update the rewards received by the suggested action to be used the next time the student uses the system. In addition, an indirect measure is used which includes the scores of the exams and assignments she received. A negative reward is added to the suggested sequence of actions throughout the semester, if the obtained final grade (in points) is decreased. For example, if the student's previous grade is 2.5, and if the new grade is 2, then this will correspond to –0.5 to be assigned as the final reward received by the set of suggested actions. If this list includes 5 suggested actions, then each one will be assigned a negative value of 0.1, which is the average value. Thus, the main goal of the system is to learn the set of actions for each student's state that will maximize her satisfaction and interactivity with the system during the semester, and at the same time enhances her learning outcomes. In the fifth step, the new state of the student is identified. This is to be done by letting the student choose between the available list of states. She is also asked about proposing a new state to be added to the system if she thinks that none of the provided states can describe her current state. This step is done for the sake of enhancing the performance of the system where the newly-suggested states will be analyzed by an expert and the list of suggested actions for the newly-added states will added to the system. This process is done offline and every while (e.g., at the end of the semester). The main challenge in personalized learning systems using RL is how to determine the State-Action-Reward triplets. In the following subsections, the three main triplets of the proposed RL-based framework will be described in more detail.

3.2 State-Action-Reward

A significant initial stage of constructing a personalized learning system is the selection of appropriate factors that should be considered and represented. The personalization is accomplished efficiently by measuring these factors. In order to determine what factors to be included when designing an effective personalized learning system, a careful and comprehensive investigation of the studies that highlight the factors that affect learning in different settings was performed. Based on this investigation, the factors to be measured can be classified into the following categories.

- Personal Factors
- Social Factors
- Cognitive Factors
- Structural Factors
- Environmental Factors

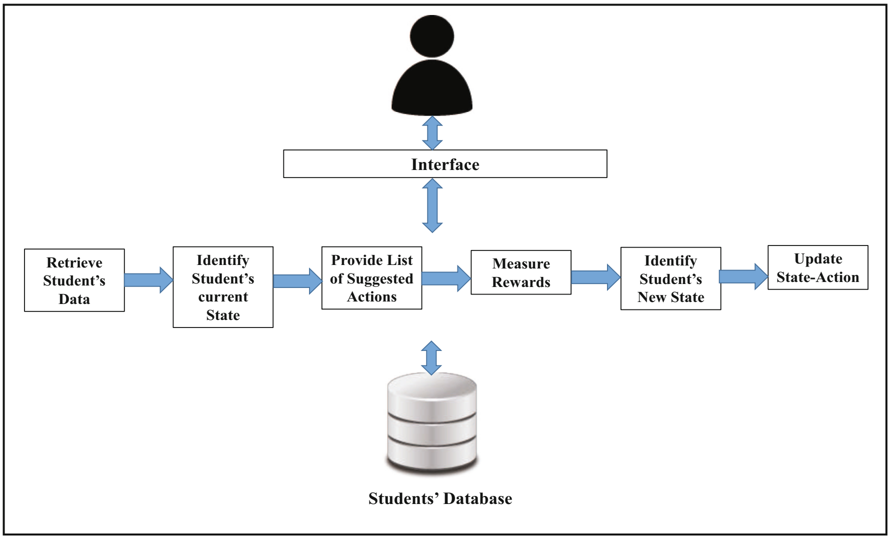
Interface
Retrieve Student's Data
Identify Student's current State
Provide List of Suggested Actions
Measure Rewards
Identify Student's New State
Update State-Action
Students' Database

Fig. 2. The framework of the proposed system

Some of the factors above are individual-level factors and others are group-level factors that should be considered when the learning setting is a collaborative one. Another possible classification for the factors to be considered is as static and dynamic factors. For example, the student's characteristics that are static include email, age, native language. Meanwhile dynamic characteristics are defined and updated each time the student interacts with the system. Some of the factors (the static ones) are set by the student at the beginning of the learning process, while the dynamic ones are usually measured through questionnaires.

Therefore, the challenge is to define the dynamic student's characteristics that constitute the base for the system's adaptation to each individual student's needs.

In the proposed approach, states are represented as a vector X = (x1, x2,…, xn), where *n* is the number of dimensions of each state. Table 1 shows the dimensions of each state with some descriptions on how these states are calculated. In addition, the table indicates a list of suggested initial actions for each possible state. Thus, when the tool is invoked, a vector attached to each learner is populated based on the values measured for each dimension. There is a large number of State-Action pairs, which makes exploring the space of possible actions very expensive. In addition, for lack of space, only a subset of possible state-action pairs are indicated in Table 1. The reward attributed towards each successful action suggestion is measured by two factors; first, the acceptance level as received from the user, second, the long term reward which represents the enhancement in the GPA.

Table 1. State-action pairs

Dimension	Possible levels	Action(s)	How the value is measured
Personality traits	Openness	Stimulate reflective learning styles by linking concepts to real life examples	Questionnaire [15]
	Conscientious	Minimal scaffolding is needed in this case Randomly provide any related learning material	
	Agreeableness	Stimulate conscientiousness by assigning small regular quizzes	
	Neuroticism	Maximal scaffolding Provide enjoyable learning to decrease anxiety Provide ways for organizing information into meaningful units. Remove test anxiety by raising self-esteem and worthiness (e.g., quotes)	
	Extraversion	Chatting and discussion with the "more knowledgeable" colleagues	
Learning styles	Activists	Learning activity need to include projects	Questionnaire [15]
	Reflectors	Explain theory using personal life examples Refer to relevant current events Use hierarchal concepts Provide affordances for summarization	
	Theorists	Ask her to organize the sequence of her thoughts	
	Pragmatists	Provide search tools Provide concept maps Ask her to write algorithms and action plans	
	Auditory	Provide audio or video lessons	

(*continued*)

Table 1. (*continued*)

Dimension	Possible levels	Action(s)	How the value is measured
	Language Visual	Provide graphical illustrations of numbers	
	Language Auditory	Provide oral explanations and numbers Use games and puzzles	
	Numerical Visual	Provide graphical illustrations of numbers	
	Visual-kinesthetic combination	Suggest experiment with self-involvement	
Prior educational achievements	GPA scores in related subjects	More scaffolding is provided for low achievements	Calculated
Intellectual skills	IQ values	More scaffolding for low IQ values	Questionnaire [16]
Perceived satisfaction about the program	5 point likert scale	Provide resources on program's objectives. Highlight and resolve the main reasons for the low satisfaction by top-management	Questionnaire [17]
Motivation	High/low	Motivate peer-peer interactions and communications with those who have high motivation measures	Questionnaire [18]
Social capital	High/low	Provide material that would motivate social presence [19]	By measuring Interactions [19]
Team-related factors: mutually shared cognition, psychological safety, cohesion, potency, and interdependence [20]	High/low	Group students with shared cognitive levels and high cohesion	Responses rates of group members, and interaction between them
Teacher-oriented factors: familiarity with the tool and beliefs	High/low	Provide teachers with instructional guidelines to increase their level of tool acceptance	Questionnaire [21]
Environment-related factors: time poorness, lighting, temperature, noise [22]	Suitable/needs adjustments	Adjust environmental factors to acceptable levels Provide automatic reminders of tasks and assignments deadlines	Sensors or feedback

4 Evaluations

The performance of the proposed framework is evaluated through simulation. In the simulation experiments, a system with 20 states and 20 actions is used. Thus, a state-action matrix is of dimensions 20 × 20. Moreover, the matrix is initially populated with randomly generated Q-values (rewards) that follow the Normal distribution (with mean = 0, and standard deviation = 1). In addition, an ε-greedy approach is used to select the action to be selected with ε set to 0.1. In an ε-greedy policy, actions with maximum rewards are selected with a probability of ε. This allows for exploring the environment, by not necessarily selecting the actions with maximum rewards. The rewards assigned to the 20 available actions for each state are randomly generated. However, for 10 of the available actions, the assigned rewards were negative, while the other 10 actions were assigned positive rewards. The learning rate is set to 0.1, together with other model's parameters. Moreover, the behaviors of 10 students were simulated. A maximum number of 100 iterations were used. Figure 3 shows the number of actions that received positive rewards for each simulated student behavior (denoted by S1 to S10 in the legend) versus the total number of iterations. As shown in the figure, the number of suggested actions that receive positive rewards increases, as the number of iterations increases. This indicates that the simulated system is able to find the best actions to be followed for each student-state pairs after a sufficient number of runs.

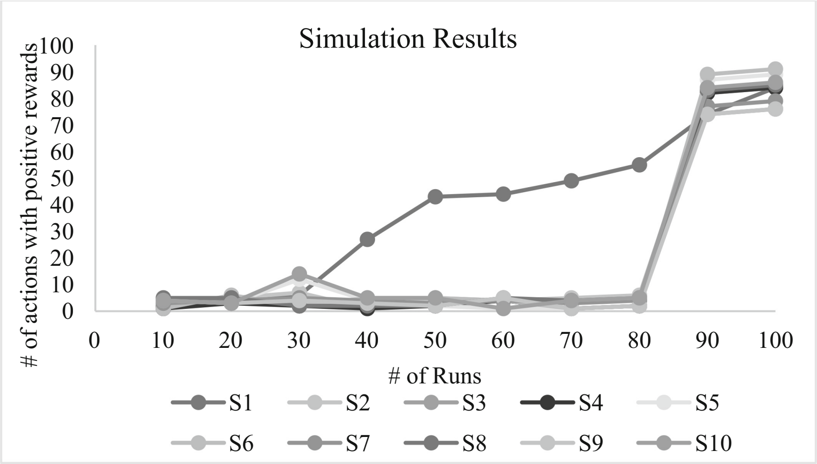

Fig. 3. Simulation results for 20 × 20 state-action matrix for 10 simulated students

5 Conclusions and Future Work

This paper presents a personalized learning framework based on RL. The proposed approach can assist the students to find out what she or he really needs, by investigating the features of a learning material or a sequence that has not been explored before. It also allows for adding the newly-suggested learning sequences by the students and/or the teachers. By investigating the history of state-action-reward for each student, the system will intelligently be able to propose the best learning environment for each student.

As a future work, it is important to add as many actions as possible for each state to allow for the exploration of the optimal one for each student-state pair. The main problem, however, is the large number of possible states or state-action values. This might cause complexity and convergence problems, especially if the model is to be implemented online without the benefit of a repetitive training period. Thus, scalability issues need to be considered in the future work.

References

1. McCarthy, B., et al.: Journey to Personalized Learning (2017)
2. Twyman, J.S.: Competency-Based Education: Supporting Personalized Learning. Connect: Making Learning Personal. Center on Innovations in Learning, Temple University (2014)
3. Shawky, D., Badawi, A., Said, T., Hozayin, R.: Affordances of computer-supported collaborative learning platforms: a systematic review. In: 2014 International Conference on Interactive Collaborative Learning (ICL), pp. 633–651. IEEE, December 2014
4. Fahmy, A., Said, Y., Shawky, D., Badawi, A.: Collaborate-it: a tool for promoting knowledge building in face-to-face collaborative learning. In: 2016 15th International Conference on Information Technology Based Higher Education and Training (ITHET), pp. 1–6. IEEE, September 2016
5. Ashraf, B., Doaa, S.: The need for a paradigm shift in CSCL. In: The Computing Conference 2017. IEEE, London (2017)
6. Said, T., Shawky, D., Badawi, A.: Identifying knowledge-building phases in computer-supported collaborative learning: a review. In: 2015 International Conference on Interactive Collaborative Learning (ICL), pp. 608–614. IEEE (2015)
7. Taraman, S., et al.: Employing Game theory and Multilevel Analysis to Predict the Factors that affect Collaborative Learning Outcomes: An Empirical Study. arXiv preprint arXiv: 1610.05075 (2017)
8. Ouf, S., et al.: A proposed paradigm for smart learning environment based on semantic web. Comput. Hum. Behav. **72**, 796–818 (2017)
9. Christudas, B.C.L., Kirubakaran, E., Thangaiah, P.R.J.: An evolutionary approach for personalization of content delivery in e-learning systems based on learner behavior forcing compatibility of learning materials. Telemat. Inform. (2017)
10. Garrido, A., Morales, L., Serina, I.: On the use of case-based planning for e-learning personalization. Expert Syst. Appl. **60**, 1–15 (2016)
11. Chrysafiadi, K., Virvou, M.: Student modeling for personalized education: a review of the literature. In: Advances in Personalized Web-Based Education, pp. 1–24. Springer, Heidelberg (2015)
12. Sutton, R.S., Barto, A.G.: Reinforcement Learning: An Introduction, vol. 1. MIT Press, Cambridge (1998)
13. Tsitsiklis, J.N., Van Roy, B.: Analysis of temporal-diffference learning with function approximation. In: Advances in neural information processing systems, pp. 1075–1081 (1997)
14. Watkins, C.J.C.H., Dayan, P.: Q-learning. Mach. Learn. **8**(3–4), 279–292 (1992)

15. Vasileva-Stojanovska, T., et al.: Impact of satisfaction, personality and learning style on educational outcomes in a blended learning environment. Learn. Individ. Differ. **38**, 127–135 (2015)
16. Chamorro-Premuzic, T., Furnham, A.: Personality, intelligence and approaches to learning as predictors of academic performance. Personal. Individ. Differ. **44**(7), 1596–1603 (2008)
17. Eom, S.B., Wen, H.J., Ashill, N.: The determinants of students' perceived learning outcomes and satisfaction in university online education: an empirical investigation. Decis. Sci. J. Innov. Educ. **4**(2), 215–235 (2006)
18. Gagne, R.M.: Learning outcomes and their effects: useful categories of human performance. Am. Psychol. **39**(4), 377 (1984)
19. Dika, S.L., Singh, K.: Applications of social capital in educational literature: a critical synthesis. Rev. Educ. Res. **72**(1), 31–60 (2002)
20. Van den Bossche, P., et al.: Social and cognitive factors driving teamwork in collaborative learning environments: team learning beliefs and behaviors. Small Group Res. **37**(5), 490–521 (2006)
21. Song, Y., Looi, C.-K.: Linking teacher beliefs, practices and student inquiry-based learning in a CSCL environment: a tale of two teachers. Int. J. Comput. Support. Collab. Learn. **7**(1), 129–159 (2012)
22. Clark, H.: Building Education: The Role of the Physical Environment in Enhancing Teaching and Research. Issues in Practice. ERIC, London (2002)