Wireless Personal Communications
https://doi.org/10.1007/s11277-020-07199-0

Reinforcement Learning Based on Contextual Bandits for Personalized Online Learning Recommendation Systems

Wacharawan Intayoad[1] · **Chayapol Kamyod**[1] · **Punnarumol Temdee**[1]

Abstract
Personalized online learning has been significantly adopted in recent years and become a potential instructional strategy in online learning. The promising way to provide personalized online learning is personalized recommendation by navigating students to suitable learning contents at the right time. However, this is a nontrivial problem as the learning environments are considered as a high degree of flexibility as students independently learn according to their characteristics, and situations. Existing recommendation methods do not work effectively in such environment. Therefore, our objective of this study is to provide personalized dynamic and continuous recommendation for online learning systems. We propose the method that is based on the contextual bandits and reinforcement learning problems which work effectively in a dynamic environment. Moreover, we propose to use the past student behaviors and current student state as the contextual information to create the policy for the reinforcement agent to make the optimal decision. We deploy real data from an online learning system to evaluate our proposed method. The proposed method is compared with the well-known methods in reinforcement learning problems, i.e. ε-greedy, greedy optimistic initial value, and upper bound confidence methods. The results depict that our proposed method significantly performs better than those benchmarking methods in our case test.

Keywords Reinforcement learning · Personalized learning · Recommendation

1 Introduction

For better online-learning experiences, the content delivery needs to take into account the learner changing context. As a result, online learning systems have been studied intensively on personalized learning by delivering suitable learning objects (LOs) to learners at the

✉ Punnarumol Temdee
punnarumol@mfu.ac.th

Wacharawan Intayoad
wacharawan.int@mfu.ac.th

Chayapol Kamyod
chayapol.kam@mfu.ac.th

[1] Computer and Communication Engineering for Capacity Building Research Unit, School of Information Technology, Mae Fah Luang University, Chiang Rai 57100, Thailand

Published online: 10 February 2020

right time. Personalized learning aims to provide content and knowledge in a way that responds to user preferences such as individual learners' objectives, learning styles and characteristic. User preferences are different from each other according to their characteristics and environment around them [14]. Personalized learning has been proven the positive effects on students [19] as it has been significantly adopted in recent years and become a potential instructional strategy to increase motivation and student success in online learning environments [20].

There are many studies on content personalization and dynamic adaption for enhancing personalized learning. A promising approach for personalized learning is the recommendation. Recommendation has become a desirable feature for online learning to support students' learning process. Providing or recommending optimal learning path tailoring to the context of the learners could enhance student learning outcome in online learning systems [2, 5]. Recommendation system in online learning is considered as interactive teaching driven by information communication technology. The general recommendation systems goal is to assist users navigation to the information that they might be interested [1, 11]. It supports individual learners by customizing a course or content delivery determining leaners' characteristics, and needs [4, 20]. In this way, an individual leaner can engage more effectively in the learning process by receiving feedback and useful information.

Recommendation for online learning typically involves three main processes: selecting relevant context information, learner classification (by the relevant contextual information) and adaptive learning methods. Selecting relevant context involves user profiles, sequential patterns of action, and information of LOs. Afterward, the contextual information is used for learner classification. The methods of the classification may be based on learning styles theories [12, 16] or static algorithms and classification methods [9, 23]. The last process of recommendation is selecting methods in order to choose which resources would be appropriate for the individual. Online learning systems use different recommendation and adaptation methods in order to suggest online learning contents and activities based on learner styles, preferences, knowledge and browsing activities.

There have been extensive studies on recommendation methods which involves collaborative filtering technique and content-based technique [10]. Collaborative filtering is based on item rating and user data. For example, knowledge level and learning style can be leveraged to provide the sequence of learning [7]. For the content-based filtering, it recommends a new item to a user based on existing users' past activities. The new recommended items always similar to the items previously taken by the user [18]. For example, the recommendation is based on similar items that users like or user profile [5, 6].

In online learning scenarios, learners' learning styles and knowledge level evolve over time. The existing methods which rely on stable indicators are not effective considering real-time adaption. In fact, online learning systems could have a flexible ability to recommend suitable contents based on the current context of learners. In addition, in online learning systems often that learners are likely new with no information about past activities, as well as for the contents in online learning, the new contents gradually added into the systems. This situation called cold-start situation [21]. To acquire such information may be expensive and there may be result in reducing user satisfaction [8].

Reinforcement learning (RL) is a well-known solution for the aforementioned issues. It is a learning algorithm that offers optimal decision making for flexible and complex environments. Differ from other recommendation approaches, RL uses reward signs instead of supervisors. The agent in RL makes optimal sequence decision based on the rewards from trial-and-error to maximize the accumulation reward. It explores the environment to get information and exploit the information to make a decision or prediction [22]. The agent

observes the environment to find the optimal action at a given state. Thereby, it is able to provide flexible adaptions in real-time based on the environment' s behavior [13]. This means that it is able to support personalized learning in which each student evolves his/her characteristic and learning behavior over time. Moreover, as RL is based on signal rewards, the agent is able to learn on the job when it does not have any information from the environment. This makes the agent able to explore when there is a new variable in the environment. Simultaneously, the agent also reasoning and learning from past experience or current knowledge to make optimal decisions. Therefore, with the learning methods of RL that is designed to act dynamically and continuously in action space, we can achieve personalized learning in online learning vision by providing suitable learning paths for learners dynamically.

To address these three challenges in the recommendation in online learning systems, we propose a method to serve the learners in their learning process. Our objective is to guide learners to the right LOs at the right time. The agent' s goal is to maximize accumulation reward by making recommendations that elicit desired user behavior (users' click as many as possible). The proposed method is based on contextual bandits as it is capable to learn in complex behavior of learners based on contextual information of learners. Unlike other previous contextual bandits methods in recommendation systems that they focus on recommending the new item or the item that users might be interested in. Our work focuses on sequential recommendation as the learning process are the crucial factor of learning achievement. We use two types of contextual information. The first one represents the past students' learning behaviors and the second one infers the current situation/state of the student. The proposed method is useful for sequentially and adaptively navigating individual students to the right content in order to motivate students and make students more engaged in the online learning processes.

The rest of the paper is organized as follows. The Sect. 2 describes the problem formulation and basic concept of RL. The section first demonstrates the full RL problem, and n-armed bandit problem and then describes the reasons why contextual bandit can be extended for our problem instead of using full RL and n-armed bandit problems. The Sect. 3 presents our proposed method and the contextual information that we use in the study. Section 4 depicts our experimental setting and results. Finally, Sect. 5 draws a conclusion and discussion.

2 Problem Formulation and Basic Concept

In this section, we first describe the basic concept of full RL problem and simple RL setting (n-armed bandit) as both settings related to our proposed method that based on contextual bandits problem. We follow notation and terminology based on Sutton and Barto [22]. Some important notations are depicted in Table 1.

2.1 Reinforcement Learning Problems

The full RL problem involves learning by interacting with the environment and aware of how the environment reacting back. RL algorithms actively explore their environments to get useful information about cause and effect, the consequences of actions, and what to do to achieve the goal [22]. The RL algorithms exploit the current information from the interaction to make decision or prediction. The goal of RL problems is to maximize accumulative reward. Reward

Table 1 Notations and descriptions

Notation	Description
s	State
a	Action in the case of full RL, and arm in the case of bandits problem
$Q_t(a)$	Estimated action-value of action a
$q(a^*)$	Optimal value of action-value
S	Set of all states
$A(s)$	Set of all possible actions in state s
R	Set of possible rewards
t	Discrete time step
T	Final time step of an episode
S_t	State at t
A_t	Action at t
R_t	Reward at t
G_t	Cumulative reward following t
π	Policy, decision making
$\pi(s)$	Action taken in state s under deterministic policy π
$V(s)$	Value function of state s
X	Set of contextual information
$X(t)$	Context at time t
γ	Discount-rate parameter

in RL problems is a scalar feedback signal to determine how well of the agent's actions. Action may take long term consequence, as a result, rewards may be delayed. In some cases may be better to sacrifice the immediate reward to get more in the long run.

The agent gets the reward from the observed environment related to the agent' s action and environment' s state that define a Markov Decision Process (MDP). The action at the given state can have both the direct effect on reward and the indirect effect on the transition of the environment' s state. These interactions between the agent and the environment occur continually. The environment responses to those actions and presents new states at each discrete time step of the action to the agent. If the sequence of the rewards from agent sequential actions from time step t is denoted to $R_t, R_{t+1}, R_{t+2}, \ldots$, and G_t is defined as a function of the reward sequence. The return of sum reward from time step t is defined as:

$$G_t = R_t + R_{t+1} + R_{t+2} + \ldots + R_T, \tag{1}$$

where T is a final time step. The agent follows the optimal policy in order to maximize the accumulative rewards. In order to determine long term expected return of any states $s \in S$, where S be a set of a finite set of states, the value function of Markov process is used. The value function of state s is defined as:

$$V(s) = E[G_t | S_t = s] \tag{2}$$

$V(s)$ can be decomposed into two parts: (1) immediate reward R_t, and (2) discount value $\gamma, (0\gamma 1)$ of the successor state $V(S_{t+1})$.

$$V(s) = E[R_t + \gamma V(S_{t+1}] \tag{3}$$

The discount rate γ impacts on lowering of future rewards based on the current state S_t. The discount rate is used to cope with the uncertainty of the real world and infinite Markov. Correspondingly, it determines the value of future rewards. If the discount rate γ is close to 0, the concentrate of the reward is based on the immediate reward.

$$V(s) = E[R_t + \gamma V(S)_{t+1}] \tag{4}$$

Instead of considering solely on the state of the environment, the value function can be estimated by determining a value of action a in a state s. The state-action function is defined as:

$$Q(s,a) = E\left[\sum_{k=0}^{\infty} \gamma^k R_{(t+k+1)} | S_t = s, A_t = a\right] \tag{5}$$

From our previous study [15], we applied Sate-Action-Reward-State-Action (SARSA) algorithm that is based on the on-policy learning method and MDP. The result from the experiments depicted that online learning system is considered as a high degree of uncertainty in the environment. The accumulative reward is optimal when we focus on the most recent reward [15], that is desirable in a non-stationary environment [22]. In addition, the actions at the given states do not have a great effect on the succession state and the reward. Therefore, in a high degree of uncertainty environment, it may not necessary to implement the full RL problem but rather focuses on a simple setting and effective. The next section introduces contextual bandits (which is the extension of n-armed bandit problem) that is applicable for recommendation systems as it focuses on the most recent action and involves in learning policy associating each task.

2.2 Contextual Bandit

Contextual bandit tasks are intermediate between n-armed bandit problems and full RL problem. It focuses on immediate rewards like n-armed bandit problems. In the same time, it involves full RL problems as it uses policy for selecting choices/actions by determining contextual information. Noted that, in n-armed Bandit problems, the term "learner" and "algorithm" are used to refer to the act on behalf of the algorithms. In this paper, we use the term "agent" that is used in full RL problem, instead of leaner.

Contextual bandit is a variant of n-armed bandit problem which is the simplified setting of RL problems. It avoids the complexity of the full RL as the agent learns to act in one situation with n different actions/options. After each choice of actions, the agent gets the payoff/reward from the environment as a numerical reward. Contextual bandit is extended by using contextual information to explore uncertain situations more effectively. We have viewed n-armed bandit problem as one state or one-step-decision-making problems. By integrating contextual information, we can view as sequential-decision-making problems that there is an information X at each step. Thereby, each action is a transition to new information $X^{'}$, with probability, $P^{A}_{X,X'}$.

The objective of the agent is as same as in the full RL problem as it focuses on maximizing the expected total reward over some period of time, i.e. over 1000 time steps of actions. Each action has an expected/mean reward. The action-value can be viewed as a stochastic problem. As a result, we determine the action-value as estimating. One of the natural ways to do is estimating by averaging the reward received when the action was selected. We denote the true action-value a as $q(a)$, and the estimated action-value of action a on the t^{th}

time step as $Q_t(a)$, where $N_t(a)$ is the number of time that action a has been chosen prior to t, yielding rewards $R_1, R_2, \cdots, R_{n_t(a)}$. The estimated action-value of action a is defined as:

$$Q_t(a) = \frac{R_1 + R_2 + \cdots + R_{n_t(a)}}{N_t(a)} \tag{6}$$

If $N_t(a) = 0$, we may set some default value to $Q_t(a)$. As $N_t(a) \to \infty$, $Q_t(a)$ converges to $q(a)$ by the law of large numbers. Such that an the optimal value of action-value is

$$q(a^*) = argmax_{a \in A} q(a) \tag{7}$$

2.3 Trail-and-Error Methods

Recall, agent objective is to maximize accumulative reward, the agent may act greedy by selecting an action $Q_t(a^*)$ based on current knowledge. However, we do not know which one is the best option, particularly in an uncertain situation. Exploration is useful in order to gain information and new knowledge. Moreover, in a high degree of uncertainty environment, the new information is higher. One of the challenges in RL problem is the trade-off between exploration and exploitation (exr/exp). RL algorithm uses information to determine which action to take rather than instruct by correct actions. In order to obtain the maximum reward, the agent prefers to take an action that likely to return the optimal reward by determining the experiences from the past. However, by acting greedy at all time, the agent will not learn any new information. Correspondingly, to discover the optimal decision, the agent learns from trial-and-error. Thereby, the agent has to exploit to obtain the expected optimal reward, but it also has to explore new actions to learn which action return optimal reward in the future. There are various methods to cope with exr/exp dilemma. We use well-known methods as the benchmarks for evaluating our proposed method later in the experiments and results section. These methods are: (1) ε-greedy, (2) optimistic initial value, (3) Upper-confidence bound (UCB).

- *ε-greedy* Greedy method always exploits the current knowledge in order to maximize cumulative reward. A simple alternative is to act greedy most of the time but once in a while randomly selects other alternative actions with the same probability distribution independence from action-value. The probability to select action randomly in ε-greedy is represent by ε. With probability ε, the actions are selected randomly.
- *Optimistic initial value* This method is different from ε-greedy as it assigns the initial value $Q_t(a)$ for all actions/arms. Instead of relying on the initial estimated reward, it set $Q_t(a)$ value to be greater than the mean reward. For instance, if the mean reward is 1, we may set the $Q_t(a) = 5$. In this way, at the beginning, the reward from any actions will be lower than $Q_t(a)$, then all the actions are tried several times before estimated reward converges.
- *UCB* It adopts the idea of ε-greedy for exploration. However, instead of randomly selecting any action with the same distribution probability, UCB method selects actions that have the potential to be optimal or the action that achieves highest upper confidence bound. First, we estimate upper confidence for each action-value at trail t as $U_t(a)$, such that $|Q_t(a) - q(a)| < U_t(a)$. UCB selects the action that maximizes upper value. This

results to $A_t = argmax_{a \in A}(Q_t(a) + U_t(a))$, where $U_t(a) = argmax_a + c\sqrt{\frac{Int}{N_t(a)}}$, Int be the natural logarithm of t, and c controlling the degree of exploration.

It is a difficult question to choose which one is the best method. The ε-greedy method chooses randomly a small probability of ε at a time. While the UCB method chooses deterministically and explores by subtly favoring at highest upper confidence bound. And the Optimistic initial value method gives the opportunity to explore equally at the beginning of learning. In order to find out which one is the suitable method for our study, we compare them by testing with the case study in the Sect. 4.

3 Methodology

The proposed recommendation method is based on contextual bandit problem. The proposed method includes two types of contextual information associated with (1) past student learning behaviors, which involve overtime, and (2) users' current situations. The agent learns policy together with reasoning contextual information in order to make the optimal actions towards situations.

In order to simplify the environment, we assume that our RL problem is a stationary problem. Learning sequences of students are viewed as an episode and each action in the episode is viewed as a trait or a step. The episode ends when the last LO is selected from the target student. Furthermore, the state-space of the environment is limited and the initial states of all learning sequences are the same.

Figure 1 depicts the framework of our proposed method for online learning recommendation. As same as RL problem, our framework is composed of two main parts. The first part is the online learning environment and the second part is the RL agent. The agent

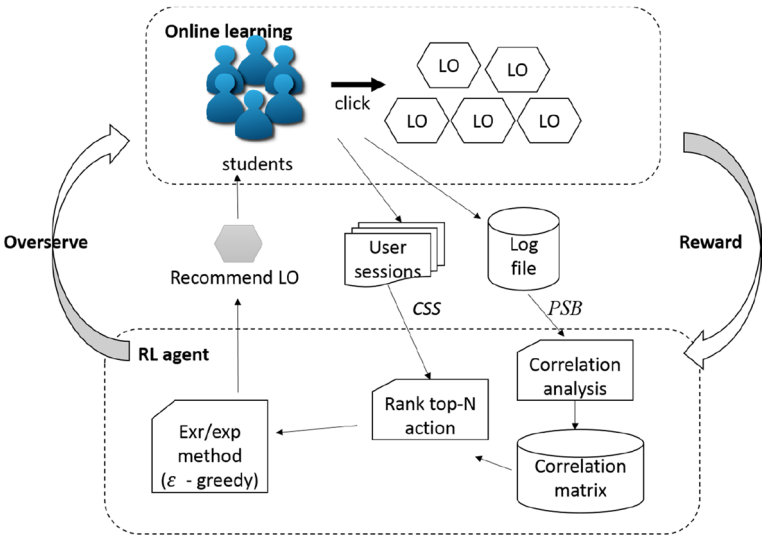

Fig. 1 The framework of the proposed contextual bandits methods for an online learning recommendation system

observes the environment to recommend suitable LOs to a target student. The framework can start where the student logs into the online learning system, then the agent will provide a recommendation. If the student selects the recommended LO, the agent will receive a numerical reward; otherwise nothing. However, the reward and penalty value can be changed. Each interaction between students and LOs is recorded in the log file. The log file represents past student's behaviors (PSB). This PSB is contextual information that is used for correlation analysis that involves the relationship between current state and actions. The result of correlation analysis stores in the correlation matrix. In order to recommend next LO, agent analyzes the current online student situation/state (CSS) and correlation matrix to provide the set of LOs which have the top highest correlation value (top-N rank action). Then, the top-N rank actions are determined to select the most optimal action-values. In our framework, we used ε-greedy for balancing the exr/exp as it is a straight forward method and simple but guarantees that the agent visits all states and all actions with the probability ε. Hence, ε-greedy is suitable for online learning systems which are considered as a high degree of uncertainty environment.

3.1 Contextual Information

For making appropriate condition criteria based on contextual information, our study used two types of contextual information for contextual bandit problems.

1. *Past student behaviors (PSB)* Each student has a different characteristic and has a different way of learning. The learning paths in online learning of each student represent navigation learning behaviors. Each click on any LO from students is recorded in the log file. We use the traces of sequential learning paths as a past experience to create the decision rules for the agent in order to recommend the right LO to students. For example, a student has a trace of the learning path as a set of LOs: $\langle a \rightarrow b \rightarrow c \rightarrow d \rightarrow e \rangle$. It means that the student started learning on LO a first and continue with LO b, c, d, and e, respectively.
2. *Current student state (CSS)* We use the target student' s session as contextual information for representing the student states and actions that have already taken in the previous trials for determining his/her current situation. This contextual information impacts action a transition to a new context X'. Thereby, the agent can recommend next LO in real-time based on the current context of the student.

3.2 Correlation Analysis

PSB is used as the contextual information for the analysis. The concept of correlation analysis is to discover pairs of actions that commonly follow one another. The more frequently observed the closer correlation between actions.

Definition 1 (*followed relation*) Let L be an log file over A, where A be a set of actions, σ be a trace (PSB) in L, and a, $b \in A$. Action a is followed by action b, $a \rightarrow b$, if :

- there is a trace $= t_1, t_2, \ldots, t_n, \sigma \in L$
- and $i \in 1, \ldots, n-1$ and $i < j \leq n$
- such that $t_i = a$ and $t_j = b$

Considering for instance $L = [\sigma_1\langle a, b, c, d\rangle, \sigma_2\langle a, c, b, d\rangle, \sigma_3\langle a, c, d\rangle]$. For this L complying with the followed relation can be found the number of times that one action followed by another one action$| \rightarrow | : |a \rightarrow b| = 1, |a \rightarrow c| = 2, |b \rightarrow c| = 1, |b \rightarrow d| = 1, |c \rightarrow b| = 1,$ and $|c \rightarrow d| = 2$. Then the result is captured in the frequency matrix of followed relation for each pair of actions. Table 2 presented the number of time one activity is followed by another activity of the log file L.

In order to measure the correlation between actions, we introduce the concept of correlation measurement which is a conditional probability that b occurs in the transaction after that a has occurred in that transaction.

Definition 2 (*correlation measurement*) Let L be an log file over A, where A be a set of actions and $a, b \in A$. The correlation of action a and b is defined by:

$$cor(a, b) = \frac{|a \rightarrow b|}{N(a)} \tag{8}$$

where $|a \rightarrow b|$ is the total number of traces that a is followed by b, and $N(a)$ is the total number of traces that contain a in L. The more value of correlation the more significant that two actions are correlated. If $cor(a, b)$ is close to 1, then there is a strong positive correlation that a is followed by b. According to Table 2, $cor(a, b) = \frac{1}{2}$.

3.3 Rank Top-*N* Recommended Actions

After correlation analysis, the CSS is used as the contextual information to predict the next action by determining the top N highest value of correlation given the current state, where N is the number of desired candidate actions. The top-N actions, $top(A_t)$, is a set of actions that are generated as the candidates to be recommended to the target user at time step t.

$$top(A_t) = (a_1, a_2, \ldots, a_N) \tag{9}$$

where $(a_1, a_2, \ldots, a_N) \in A$, but not in the previous taken actions of the target student, $top(A_t) \neq (a_{t=1}, a_{t=2}, \ldots, a_{t-1})$, and n be a number of the desired top actions for recommended $top(A_t)$ to the target student at time t. The agent does not take into account the duplicated actions that already taken by the target student. As agent only determines the learning path and assumes that the target student already achieves the learning objects from the previous LOs.

For example, given target student' s active session $\langle d, b, c\rangle$, the $cor(c, x_i)$ values are sorted, where $x_i \in A$ and x_i is not in $\langle d, b\rangle$.

Table 2 Correlation matrix of followed relation in L

$\| \rightarrow \|$	a	b	c	d
a	0	1	1	0
b	0	0	1	1
c	0	1	0	2
d	0	0	0	0

3.4 Recommended Action

Recall that the classical dilemma of RL problem is the tradeoff between exr/exp. In our study, we use a well-known and effective method ε-greedy algorithm combing with the contextual information in order to make a decision. Instead of solely determining the most optimal action value for exploitation, we use the predefined set of action $top(A_t)$ as choices.

$$a_n = \quad argmax_{qa^* \in topA_t} q(a^*) \quad if(i > \varepsilon) \tag{10a}$$

$$random() \qquad\qquad otherwise \tag{10b}$$

Let a_n be an recommended action, $q(a^*)$ be the optimal value of action-value, where $q(a^*) \in top(A_{t+1})$. $random()$ is a function for return the random item, allowing to perform exploration. i is a random value with uniformly distributed over [0,1], ε defines the exr/exp tradeoff. The more ε is the greedier. After receiving a reward, the algorithm improves its action selection strategy with the new observation.

ε-greedy is practical in our context as the in learning environment is dynamic and the goal is to learn to map the situation to the best action. However, the true action changes over time. As a result, the agent uses ε-greedy method to choose randomly a small fraction of the time. This means that ε-greedy allows the agent to learn in new situations. In this study, we present the performance of ε-greedy combing with contextual-bandits RL through the test and comparison with other methods for balancing exr/exp problem.

4 Experimental Settings and Results

Our goal is to measure the performance of the proposed method which is the rule for selecting an action at each time step based on the contextual information. In this section, we first explain our dataset. Then we introduce the evaluation method which based on click-through rate. And the last part, we depict the results of the experiments comparing with the benchmarking methods.

4.1 Data Gathering

The study uses real log file from an e-learning system. The log contains the history of the interactions between students and LOs. The log records student ID and the sequences of the LOs that students visited. There are 365 students, 2,519 events, and 78 LOs from the log file. Each student has a different background, i.e. education background, age, and gender.

As we have mentioned in the previous section that the proposed method does not recommend duplicate actions that already taken or recommended. Thereby, we remove the data items which contain duplicate student ID and LOs in order to make it simple for evaluation and the effectiveness of recommendation. For instance, student ID 01 interacted with LO1 and later on the student interacted with the same LO1 again. Only the first time that student interacted with LO1 remains in our test dataset and the rest are removed.

4.2 Evaluation Method

In the context of the online learning recommendation systems, the objective of the systems is to navigate the students to the suitable LO at the right time in order to support personalized learning for individuals. As such, we consider LOs as the armed in each state and the recommended LOs can be viewed as the action in RL problem. Each click of students over the LO is the payoff or reward from the environment to the agent. We use the term click-through rate (CTR) to represent the reward. The CTR represents the ratio between the total number of clicks and the total number of recommended actions. If a student clicks on the recommended LO a reward of 1 is incurred; otherwise, the reward is 0. Therefore, agent objective is to maximize the accumulation number of CTR. The initial setting for all $Q(a)$ is 0. As a result, the proposed method can be tested in the cold start problem.

We compare our proposed method with the n-armed bandit algorithms that ignore all contextual information. The widely used methods are the benchmarks : (1) ϵ-greedy, (2) greedy optimistic initial value, (3) and UCB [3, 17, 22].

4.3 Result

Before comparing the proposed method with the benchmark. We first observe the performance of the benchmarking methods. Figure 2 depicts the results from ϵ- greedy that has different parameter of ϵ as 0.2, 0.5, and 0.9. We calculate CTR over every 100 iterations of t. And we run the simulation until the number of t reached 12,000. The horizon axis presents the number of interactions and the vertical axis is the performance metric (average CTR). It can be observed that the convergence average CTR increases as the ϵ decreases. In long run the, $\epsilon = 0.2$ has the best performance, while $\epsilon = 0.9$ has the lowest CTR.

Thereby, we choose $\epsilon = 0.2$ to compare with other benchmarking methods and the proposed method. For greedy optimistic initial value, the default value for $q(a)$ is 2 for all arms. And the agent only uses the current knowledge for exploitation (greedy). For UCB, we set $c = 1$. The proposed method follows the optimal $\epsilon = 0.2$ from previous result of ϵ- greedy.

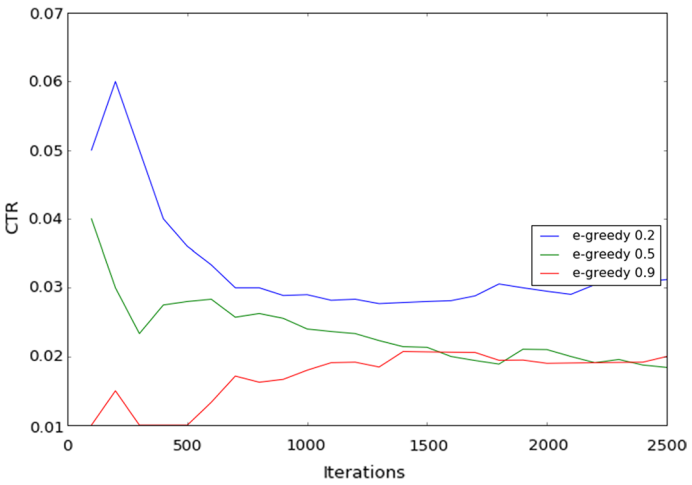

Fig. 2 Parameter tuning for ϵ-greedy

Figure 3 represents the results of our experiments. The performance of ϵ-greedy, greedy optimistic initial value, UCB and our proposed contextual bandit method are compared. Our observation is that at the beginning, all the benchmarking methods have the similar performance. And the proposed method performs better than other benchmarking methods from the early stage. After 2000 iterations, the benchmarking methods ϵ-greedy, greedy-optimistic, and UCB reach their convergence of average CTR. While our proposed method could use contextual information to explore the student state and the correlative between states and actions to improve the recommendation over time. The performance of the proposed method significantly improves the convergence of the average CTR.

We later compare the methods when data is sparse as the small scale of data is one of the most challenges in personalized web service [3, 17]. In our experiments we reduce data sizes to 30 %, 20 %, 10 %, 5 %, and 1 %, respectively to mimic the situation that we have a fixed number of students. Figure 4 depicts the comparison results with various data sparsity levels. The first observation is that at the 1 % data size, ϵ-greedy and the proposed contextual bandit method have better performance than greedy optimal and UCB. The second observation is that the proposed contextual bandit method consistently outperforms the benchmarking methods even the data size is small. Thereby, the proposed contextual bandit method is useful in the small scale of data scenario. The third observation focuses on the benchmarking methods performances. For the small data size, their performances are not identical difference. And the last observation is that all methods do not significantly benefit from a small data size.

5 Discussion

In order to provide suitable LOs for students in online learning systems, we proposed the recommendation method based on contextual bandits reinforcement learning. The proposed method is able to provide dynamically and continuously recommendation based on user learning behaviors that involved over time by determining the related contextual information. Our proposed method benefits from contextual bandit RL as it fast and simple to implement in

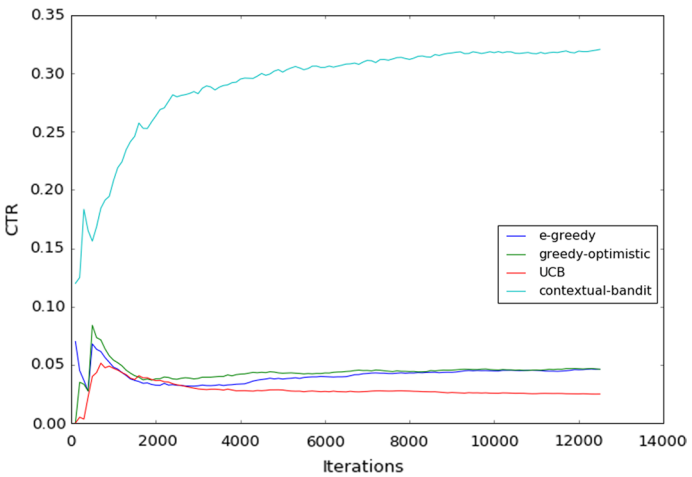

Fig. 3 Average CTR for exr/exp algorithms

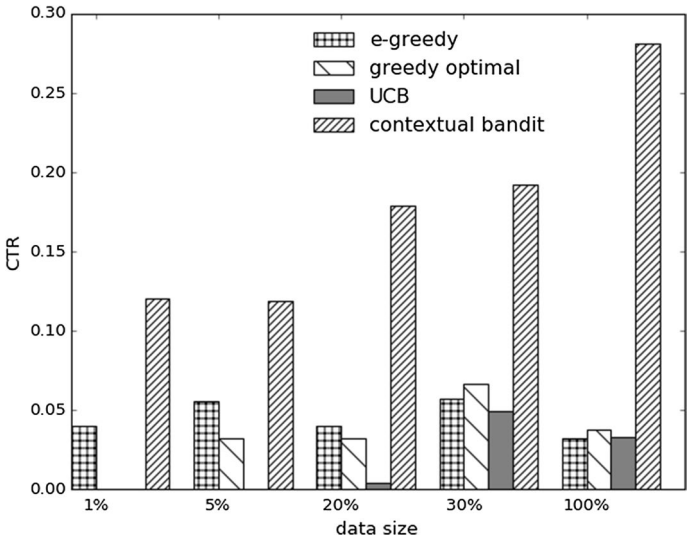

Fig. 4 Average CTR for different data size

a high degree of uncertainty environment when comparing to full RL problem. Moreover, applying contextual information allows agent to explore uncertainty situation more effectively.

The study proposed two types of contextual information, the first one is past student behaviors (PSB) and the second one is current student state(CSS). With PBS, we can discover the correlation between learning objects and student behaviors. On the other hand, CSS is used as the current contextual information inferring the current state of students. Combing these two types of contextual information, the method can provide the set of LOs that highly associate with the students learning process.

The results of the experiments demonstrated that the proposed method can effectively address the challenge in online learning recommendation. The proposed method is able to improve the optimal decision making in flexible environments. It outperformed the well-known methods such as ε-greedy, greedy with optimistic initial value, and UCB when comparing the average click-through rate. Moreover, in order to cope with the general problem of personalized web service. We tested our proposed methods when data is sparse. According to experimental results, our proposed contextual bandits method had better performance than benchmarking methods. Such that the proposed method is able to cope with the cold-start problem in personalized web service.

However, it is a general question in RL problems to ask which the best method for balancing exr/exp is. There are various sophisticated methods available. In order to decide which method going to use for RL problem depends on many factors, such as the precision value of the estimations, uncertainties, and the type of the remaining steps (stationary/non-stationary problem).

6 Conclusion

In this paper, we present a method for recommendation in online learning systems. The proposed method is based on the contextual bandit. We introduced two contexts in the proposed method: past student behaviors (PSB) and current student state (CSS). PSB is the learning path of each student representing navigation learning behaviors. An agent from RL learns PSB as a past experience to create the decision rules for recommendations. CSS refers to the student states and actions that have already taken in the previous trials. The agent determines PSB and CSS in order to recommend the right LO in real-time. In addition, we introduced exploration strategy into the proposed method to improve the engagement between students and online learning systems. Our methods can effectively make right recommendation to students as the proposed method is able to achieve the highest reward (CTR) comparing with other benchmarking methods.

In the future work, we plan to discover the relationship between other contexts, for example, the correlation among LOs in online learning environments, as the new contents often are added or integrated to the previous item for updating. And also the context information about the status of knowledge of the target students, such as test scores. As these types of context information can be used by the agent to recommend the LOs to the students who have not achieved the learning objective yet. In addition, the correlation analysis might be not the most effective way when facing non-stationary problems or the large state space. This problem may require offline maintenance or new techniques and methods to integrated or update the performance of the proposed method.

Funding Funding was provided by Mae Fah Luang University.

Compliance with Ethical Standards

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Adomavicius, G., & Tuzhilin, A. (2005). Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering*, *17*(6), 734–749.
2. Basu, P., Bhattacharya, S., & Roy, S. (2013). Online recommendation of learning path for an e-learner under virtual university. In *International conference on distributed computing and internet technology* (pp. 126–136). Springer.
3. Bouneffouf, D., Bouzeghoub, A., & Gançarski, A. L. (2012). A contextual-bandit algorithm for mobile context-aware recommender system. In *International conference on neural information processing* (pp. 324–331). Springer.
4. Brusilovsky, P., & Peylo, C. (2003). Adaptive and intelligent web-based educational systems. *International Journal of Artificial Intelligence in Education*, *13*, 159–172.
5. Chen, C. M. (2008). Intelligent web-based learning system with personalized learning path guidance. *Computers & Education*, *51*(2), 787–814. https://doi.org/10.1016/j.compedu.2007.08.004.
6. Chen, C. M., & Duh, L. J. (2008). Personalized web-based tutoring system based on fuzzy item response theory. *Expert Systems with Applications*, *34*(4), 2298–2315. https://doi.org/10.1016/j.eswa.2007.03.010.
7. Chrysafiadi, K., & Virvou, M. (2015). Fuzzy logic for adaptive instruction in an e-learning environment for computer programming. *IEEE Transactions on Fuzzy Systems*, *23*(1), 164–177.

8. Chu, W., & Park, S. T. (2009). Personalized recommendation on dynamic content using predictive bilinear models. In *Proceedings of the 18th international conference on world wide web* (pp. 691–700). ACM.
9. Dascalu, M. I., Bodea, C. N., Moldoveanu, A., Mohora, A., Lytras, M., & de Pablos, P. O. (2015). A recommender agent based on learning styles for better virtual collaborative learning experiences. *Computers in Human Behavior*, *45*, 243–253. https://doi.org/10.1016/j.chb.2014.12.027.
10. Drachsler, H., Hummel, H. G., & Koper, R. (2008). Personal recommender systems for learners in lifelong learning networks: The requirements, techniques and model. *International Journal of Learning Technology*, *3*(4), 404–423.
11. Fan, Y., Shen, Y., & Mai, J. (2008). Study of the model of e-commerce personalized recommendation system based on data mining. In *International symposium on electronic commerce and security, 2008* (pp. 647–651). IEEE.
12. Felder, R. M., & Silverman, L. K. (1988). Learning and teaching styles in engineering education. *Engineering Education*, *78*(7), 674–681.
13. Golovin, N., & Rahm, E. (2004). Reinforcement learning architecture for web recommendations. In *Null* (p. 398). IEEE.
14. Hong, J., Suh, E. H., Kim, J., & Kim, S. (2009). Context-aware system for proactive personalized service based on context history. *Expert Systems with Applications*, *36*(4), 7448–7457.
15. Intayoad, W., Kamyod, C., & Temdee, P. (2018). Reinforcement learning for online learning recommendation system. In *The 6th global wireless summit (GWS-2018)*. IEEE.
16. Kolb, D. A. (1981). Learning styles and disciplinary differences. *The Modern American College*, *1*, 232–255.
17. Li, L., Chu, W., Langford, J., & Schapire, R. E. (2010). A contextual-bandit approach to personalized news article recommendation. In *Proceedings of the 19th international conference on world wide web* (pp. 661–670). ACM.
18. Mladenic, D. (1999). Text-learning and related intelligent agents: A survey. *IEEE Intelligent Systems and Their Applications*, *14*(4), 44–54.
19. Pane, J. F., Steiner, E. D., Baird, M. D., & Hamilton, L. S. (2015). *Continued progress: Promises evidence on personalized learning*. Santa Monica: RAND Corporation.
20. Paramythis, A., & Loidl-Reisinger, S. (2003). Adaptive learning environments and e-learning standards. In *Second European conference on e-learning* (Vol. 1, pp. 369–379).
21. Park, S. T., Pennock, D., Madani, O., Good, N., & DeCoste D. (2006). Naïve filterbots for robust cold-start recommendations. In *Proceedings of the 12th ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 699–705). ACM.
22. Sutton, R. S., & Barto, A. G. (1998). *Reinforcement learning: An introduction*. Cambridge: MIT Press.
23. Tarus, J. K., Niu, Z., & Yousif, A. (2017). A hybrid knowledge-based recommender system for e-learning based on ontology and sequential pattern mining. *Future Generation Computer Systems*, *72*, 37–48. https://doi.org/10.1016/j.future.2017.02.049.

Wacharawan Intayoad She earned her BA of Management of Information Systems in 2003 from Thammasat University, Thailand and M.Sc. of Information Systems in 2008 from Lund University, Sweden. Her research interests are data analytics, big data, machine learning, artificial intelligence in education and logistics domains.

Chayapol Kamyod He received his Ph.D. in Wireless Communication from the Center of TeleInFrastruktur (CTIF) at Aalborg University (AAU), Denmark. He received M. Eng. in Electrical Engineering from The City College of New York, New York, USA. In addition, he received B.Eng. in Telecommunication Engineering and M. Sci. in Laser Technology and Photonics from Suranaree University of Technology, Nakhon Ratchasima, Thailand. He is currently a lecturer in Computer Engineering program at School of Information Technology, Mae Fah Luang University, Chiang Rai, Thailand. His research interests are resilience and reliability of computer network and system, wireless sensor networks, embedded technology, and IoT applications.

Punnarumol Temdee She received B.Eng. in Electronic and Telecommunication Engineering, M. Eng in Electrical Engineering, and Ph.D. in Electrical and Computer Engineering from King Mongkut's University of Technology Thonburi. She is currently a lecturer at School of Information Technology, Mae Fah Luang University, Chiang Rai, Thailand. Her research interests are social network analysis, artificial intelligence, software agent, context-aware computing, and ubiquitous computing.