Optimizing Adaptive Notifications in Mobile Health Interventions Systems: Reinforcement Learning from a Data-driven Behavioral Simulator

Wang, S.; Zhang, C.; Kröse, B.; van Hoof, H.

DOI
10.1007/s10916-021-01773-0

Publication date
2021

Document Version
Final published version

Published in
Journal of medical systems

Journal of Medical Systems (2021) 45: 102
https://doi.org/10.1007/s10916-021-01773-0

MOBILE & WIRELESS HEALTH

Optimizing Adaptive Notifications in Mobile Health Interventions Systems: Reinforcement Learning from a Data-driven Behavioral Simulator

Shihan Wang[1,2] · Chao Zhang[3,5] · Ben Kröse[1,4] · Herke van Hoof[1]

Received: 4 June 2021 / Accepted: 20 September 2021 / Published online: 18 October 2021

Abstract
Mobile health (mHealth) intervention systems can employ adaptive strategies to interact with users. Instead of designing such complex strategies manually, reinforcement learning (RL) can be used to adaptively optimize intervention strategies concerning the user's context. In this paper, we focus on the issue of overwhelming interactions when learning a good adaptive strategy for the user in RL-based mHealth intervention agents. We present a data-driven approach integrating psychological insights and knowledge of historical data. It allows RL agents to optimize the strategy of delivering context-aware notifications from empirical data when counterfactual information (user responses when receiving notifications) is missing. Our approach also considers a constraint on the frequency of notifications, which reduces the interaction burden for users. We evaluated our approach in several simulation scenarios using real large-scale running data. The results indicate that our RL agent can deliver notifications in a manner that realizes a higher behavioral impact than context-blind strategies.

Keywords Mobile health intervention · Adaptive agent · Reinforcement learning · Human simulator · Just-in-time adaptive intervention

Introduction

Adaptive interventions have emerged as a new perspective of prevention and treatment in healthcare [1]. The just-in-time adaptive intervention (JITAI) is an adaptive intervention design concept, aiming to provide the right type /amount of support at the right time based on an individual's changing internal and external states [2, 3]. Though JITAIs can be administered through several means (e.g. in-person and computer), the ubiquity of mobile devices allows for continuous participant monitoring and delivery of personalized interventions. Mobile health systems (agents) with JITAIs have proven effective in preventing certain health threats (e.g. overeating [4], smoking [5] and prolonged sedentary behaviors [6]) and eliciting beneficial health outcomes (e.g. increased physical activity [7] and self-management support related to chronic diseases [8]). However, the design of such interventions is demanding and the interaction with the user can be complex. Reinforcement learning (RL) based agents have been used to optimize mobile healthcare interventions adaptively [9–11], which make use of historical data or data collected on the run. The problem of historical data is that it often misses counterfactual information (i.e. what would have been the outcome had interventions or circumstances been different). The problem of data collected during the intervention is that it requires many interactions in a short period, which add burden for the user and adversely impact engagement [12, 13].

Throughout the paper, we focus on optimizing the delivery of context-aware notifications in mobile health systems. These notifications are sent in an adaptive manner dependent on the temporal and environmental context of users,

This article is part of the Topical Collection on *Cognitive Agents for Smart Health*.

✉ Shihan Wang
s.wang2@uu.nl

1 Informatics Institute, University of Amsterdam, Amsterdam, Netherlands

2 Information and Computing Sciences, Utrecht University, Utrecht, Netherlands

3 Department of Psychology, Utrecht University, Utrecht, Netherlands

4 Digital Life, Amsterdam University of Applied Sciences, Amsterdam, Netherlands

5 Human-Technology Interaction, Eindhoven University of Technology, Eindhoven, Netherlands

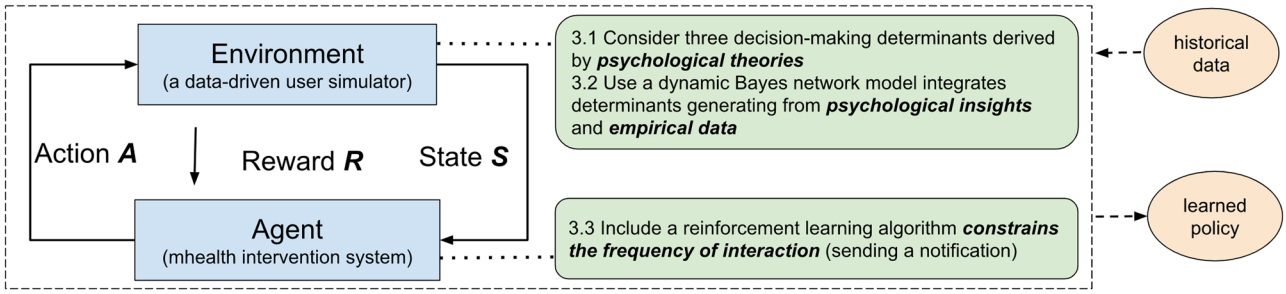

Fig. 1 The overview of our methodology, including the agent–environment interaction in the MDP model and three key components developed in both environment and agent. The approach optimizes the delivery of context-aware notifications from empirical data

motivating them to perform a target activity. To solve the two mentioned problems, based on a framework that combines historical data and psychological theories about human decision-making, we developed a simulation environment to optimize the timing of these notifications. Moreover, to restrict interaction burden, we adapted an RL algorithm by incorporating a constraint on the number of notifications that can be sent within a period. Finally, we conducted a case study on promoting running activity to demonstrate our approach. A dataset covering over 10K real users' running activity was used to build our simulator and evaluate our RL agent.

Related work

For the optimization of JITAI intervention in mHealth systems, several different strategies were taken by researchers using RL [7, 8, 10, 11, 14]. However, most of those RL approaches require the agent to interact many times with the user before performing well. To shorten the online learning process, several researchers followed the concept of transfer learning to perform faster learning in mHealth settings. Tabatabaei et al. [15] and Tomkins et al. [16] make RL algorithms quickly learn from the limited experience at the beginning stage by considering similar users. Gonul et al. [17] transfer the common knowledge acquired in other environments to get faster convergence. Without constraints on the intervention frequency, those RL approaches might still bother users by too many interventions during fast learning. While they concentrate on using data collected during the online interventions, we follow another direction to solve this challenge, i.e., incorporating prior knowledge from historical data to optimize the policy in advance. Similar to our approach, Liao et al. [18] and Ameko et al. [19] integrate prior distributions using collected data in an RL optimization process. However, they apply relative small datasets in pre-learning because experimental data for specific intervention situations are often involving user interaction and therefore expensive to collect. Our framework allows learning prior knowledge from historical data collected without interacting with users, which makes the usage of large-scale data possible. To avoid many interactions in a short period, our approach for the first time performs a structural study to incorporate a constraint on interaction frequency in RL-based mHealth systems.

Methodology

We model how users sequentially decide on whether to perform a target activity when receiving notifications (we use running as an example in this paper, in this case, the mobile agent sends notifications for promoting running activities). We formalize our problem (i.e. learning the optimal strategy for delivering notifications) as a finite horizon Markov Decision Process (MDP) [20]. Figure 1 presents an overview of our approach. Here, the agent represents a mobile system that interacts with a target user (i.e. the environment) to optimize the strategy. Our agent and environment interact in a sequence of discrete and finite time steps $\{1, 2, .., t\}$, which can be naturally broken into episodes. At each time step, the agent observes a representation of the environment and selects an action accordingly (two possible actions in our case: send a notification or not send). The environment then passes a numerical reward back to the agent. Based on this feedback mechanism, the agent adapts its policy to maximize an expected long-term reward. Since too frequent interactions with the environment are not desirable in mHealth settings, we constrained the maximum number of notifications sent in each week (i.e. episode). In this paper, our optimization goal is to wisely deliver a restricted number of notifications to maximize the user's weekly running frequency.

Insight from psychological theories

Conceptually, it can be assumed that users' decisions to engage in certain activities (e.g., running) after notifications take two steps, *option generation* and *option evaluation* [21–23]. At any decision moment, behavioral options have

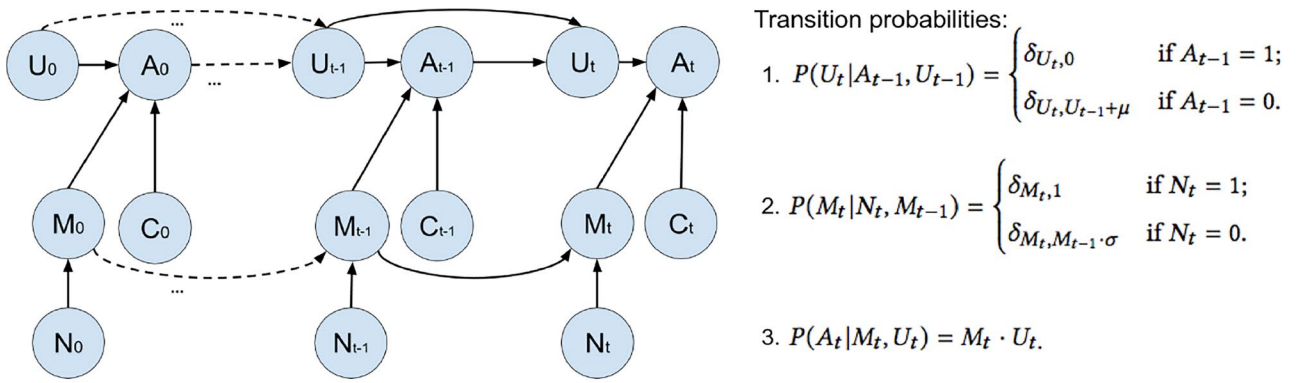

Fig. 2 Topological structure and transition probabilities of our dynamic Bayesian network

to be generated in memory before they can be compared to inform a final choice. Memory accessibility of different options during option generation is influenced by environmental cues, including system notifications. When a user receives a notification for running, the memory accessibility of running reaches its maximum. This accessibility then gradually decreases in the form of a memory decay until the next notification is received. The form of memory decay, or forgetting curve, is modeled as exponential functions in the psychology literature [22, 24].

After being generated, a target option (running) has to compete with other generated behavioral options (e.g. working on a paper) in terms of how much they satisfy a user's personal goals, such as being healthy and productive. According to classic decision-making models [25, 26], the goal-satisfying values of options, weighted by the importance of the goals, are transformed into subjective utilities, and the option with the highest subjective utility will be chosen. Without enumerating all goal-related attributes, two types of attributes are important for running behavior. First, a user's momentary context (e.g. time and weather) can have great impacts on decisions because the options' goal-satisfying values depend on the contextual variables [27]. For example, a Sunday morning with good weather makes running more enjoyable and also less interfering with one's work-related goals. Second, recently having a run ought to temporarily lower the utility of running. After a run, one's body certainly needs time to recover to a level that is sufficient for running again. Furthermore, having a run satisfies running-related goals and attenuates the importance of the goals. As people pursue multiple goals, this psychological mechanism allows people to switch to other goals and engage in behaviors that satisfy those goals (e.g. finishing a manuscript to be productive) [28]. In summary, three key determinants of running decision - *memory accessibility* of running, *urge* of running, and personal *context* - were derived from the above theories and included in our computational model.

Computational model

We formalized the above procedure as a dynamic Bayesian network (DBN). As a probabilistic graphical model, the DBN considers a set of variables and their conditional dependencies over adjacent time steps [29]. In this way, we generated a stochastic human simulator to make decisions based on both contextual and cognitive states sequentially.

Representation and topology of the DBN

Following the psychological theories above, we defined five variables and their dependencies in our DBN as follows:

- A_t represents whether an user decides to take a target activity (running) at time t.
- M_t is the user's memory accessibility of running at time t.
- U_t is the user's urge to run at time t.
- C_t is the personal context of the user at time t.
- N_t represents whether the user receives a notification at time t.

The variable M_t and U_t are real values in (0, 1). The variable N_t and A_t are binary values $\in \{0, 1\}$, where '1' represents 'receive a notification' and 'decide to run' respectively. The variable C_t includes a set of contextual features, defined as a vector of values. Under the first-order Markov assumption, we proposed a topological structure of the DBN, as shown in Fig. 2.

Definitions and inference of the DBN

We specified transition probabilities in the DBN from either empirical data or psychological insights. Based on Insight from Psychological Theories, the state transitions of U_t and M_t were defined as Eqs. 1 and 2 in Fig. 2, where the notation δ represents the Kronecker delta function [30]. Given

a certain A_{t-1} and N_t, we deterministically have U_t and M_t. The parameters μ and σ define the changing rate of *urge* and *memory accessibility*. While memory accessibility decreases exponentially, the urge to run increases linearly over time. We also defined the transition from a joint observation of M_t and U_t to a target activity A_t as Eq. 3 in Fig. 2. In particular, we proposed to calculate two probabilities $P(C_t)$ and $P(C_t|A_t)$ from empirical data (for details, see Data Description and Processing). Given these probabilities, we used the following equation to estimate how a user reacts to notifications.

$$\begin{aligned} &P(A_t|M_{0\dots t-1}, U_{0\dots t-1}, C_{0\dots t}, N_{0\dots t}) \\ &= P(A_t|C_t, N_t, A_{t-1}, M_{t-1}, U_{t-1}) \\ &= \sum_{M_t} \sum_{U_t} \frac{P(C_t|A_t)}{P(C_t)} \cdot P(A_t|U_t, M_t) \\ &\quad \cdot P(M_t|N_t, M_{t-1}) \cdot P(U_t|A_{t-1}, U_{t-1}). \end{aligned} \tag{1}$$

probability of certain discrete actions. After reaching the maximum number of notifications in each episode, the probability of sending a notification is always 0. In this way, we make sure our RL algorithm learns to deliver a restricted number of notifications according to the given momentary state.

Simulation experiments using real data

We demonstrated the performance of our approach in a case study, aiming at promoting running activities by sending context-aware notifications. Our approach was evaluated in a simulation environment using real running data.

Algorithm 1 REINFORCE with baseline & restriction

Input: initial policy $\pi(A|S,\theta)$, maximum number of notifications in each episode m

for each episode **do**
  generate $s_0, a_0, r_0, \cdots, s_{T-1}, a_{T-1}, r_{T-1}$ using θ
  if m notifications have sent at time t **then**
    let $a_t = 0$
  end if
  for $t = 1$ to $t = T-1$ **do**
    $G_t \leftarrow \sum_{k=t+1}^{T} \gamma^{k-t-1} r_k$
    $\bar{G}_t \leftarrow$ mean of G_t in the past n episodes
    if m notifications have sent at time t **then**
      continue
    else
      $\theta \leftarrow \theta + \alpha(G_t - \bar{G}_t) \bigtriangledown_\theta \log \pi(a_t|s_t, \theta)$
    end if
  end for
end for
return the learned policy $\pi(A|S,\theta)$

Reinforcement learning algorithm

To learn the optimal policy (i.e. a stochastic mapping between a personal state of the user and an action to take) in our restricted setting, we adopted a policy gradient RL algorithm, REINFORCE [31]. The REINFORCE algorithm with baseline and restriction is outlined in Algorithm 1. Our algorithm updates based on episodes. In each episode, it performs a gradient step on a neural network to optimize the policy parameter θ. We inserted a baseline function $\bar{G}_t$ inside the expectation to reduce the high variance, using the average of all returns G_t in the past n episodes. Moreover, to integrate with the restricted setting, we adjusted the procedures of action selection and policy adaptation in the REINFORCE algorithm. Inspired by clipping the continuous action space in policy gradient [32], we constrained the

Experimental data and settings

Data description and processing

We used two datasets to derive the context related distributions in Eq. 1. First, a running dataset was used to derive the distribution $P(C_t|A_t)$, measuring the relation between user context and running behavior. The data contains around 406K runs contributed by over 10K Dutch users while using a mobile fitness app from 2013-03 to 2017-03 [33]. For each run, a set of metadata is collected and timestamp and weather information at the beginning are marked. We considered six variables in the data, namely 'hour of the day', 'weekday', 'temperature', 'weather type', 'wind type' and 'humidity type'. An example of context data is *{8:00, Monday, -2, cloudy, moderate wind, moderate humidity}*.

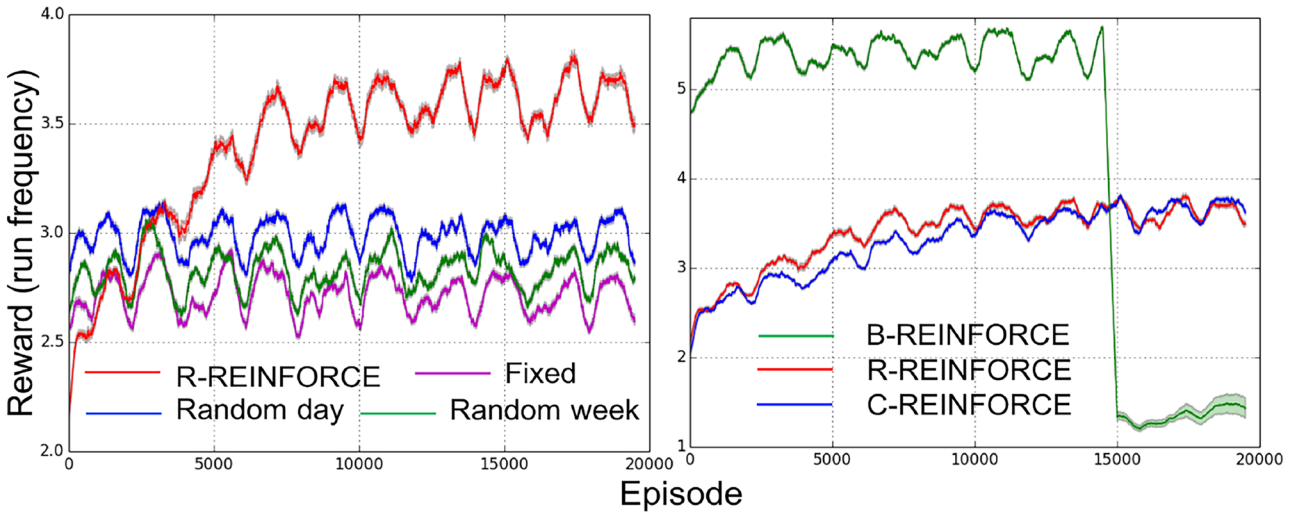

Fig. 3 The simulation results shown the average reward of agents in the sliding windows of 500 episodes

Second, an open dataset provided by the Royal Netherlands Meteorological Institute (KNMI)[1] was used to derive $P(C_t)$, the prior distribution of contextual information (general Dutch weather), which contains around 439K records of hourly weather. To make the two datasets comparable, we used the weather data over the same period of the running data.

We derived distribution $P(C_t|A_t)$ and $P(C_t)$ from the running and the weather dataset in a same manner. Thus, we only demonstrate how we derived the context distribution from the running data. Since data are only available when a running activity is performed, we concentrate on computing the distribution $P(C_t|A_t = 1)$, which is a joint distribution of all contextual variables. Since we noticed that the feature 'weekday' is conditionally independent with other features, we learned the distribution $P(\text{weekday}_t|A_t = 1)$ by computing probabilities of all seven values in the categorized feature 'weekday'. We also extracted the joint distribution of all the other features. For each combination of the discrete variables (weather, wind and humidity), we learned a separate multivariate Gaussian distribution for continuous variables (hour and temperature) using maximum likelihood estimation.

Setting of simulation with real contextual data

We implemented our simulation experiments using python[2]. The RL algorithm was developed based on pytorch[3], and our RL agent and simulation environment were built following the framework of OpenAI gym[4]. In the simulation, the agent makes a decision on whether to send notification at every hour from 8:00 to 20:00. Only when the user performs a run before the next decision time step (within one hour), the agent gets a reward of 1.0 (otherwise zero reward). In our environment, each episode is one week and maximum of 14 notifications are allowed in each week. We also provided realistic context information in the simulation environment by using empirical data in the used KNMI dataset. Based on the results of a simulator verification[5], we set memory retention rate (σ in Eq. 2) at 0.8 and urge recovery rate (μ in Eq. 1) at 0.05. The discount factor γ and learning rate α are set to 1 and 0.001 respectively. We ran each simulation 20 times. In each run, the environment starts at 0:00 of a ***random*** date with its corresponding real weather data.

Experimental results

We evaluated our data-driven RL approach in two experiments. To set a comparable environment, we randomly initialize a single simulation environment for all agents of each experiment at every simulation run.

Evaluation of context-aware policy

The first experiment aims to examine whether the policy learned by our data-driven approach outperforms general rule-based policies (not considering the contextual

[1] https://knmi.nl/nederland-nu/klimatologie/uurgegevens (last access on Oct 15th 2021)
[2] https://github.com/sw1989/RLforPAUL
[3] https://pytorch.org/
[4] https://github.com/openai/gym
[5] For details about the verification, see the supplementary information.

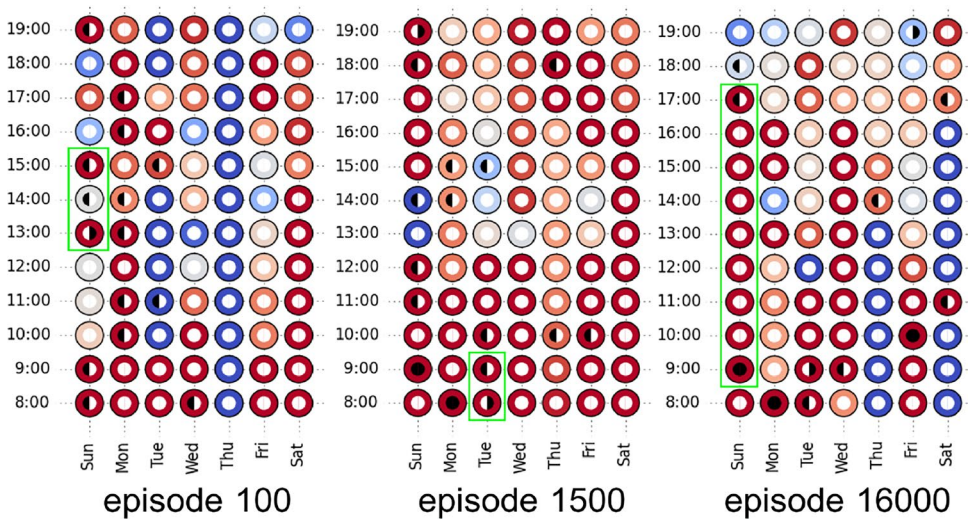

Fig. 4 Information of three episodes in the ***R*** (R-REINFORCE) agent. Each circle represents one decision point, marked by hour and weekday. Black on the left side means 'a notification', and black on the right side means 'a run'. The color of a circle represents the context desirability for running. While red and blue color correspond to the high and low desirability respectively, darker is more extreme

information of users). We compared our RL-based agent (***R*** agent) with three baseline agents. All four agents send the same number of notifications per episode, but use different strategies. Three strategies of the baseline agents are (1) 'random week agent' sends 14 notifications randomly in each week; (2) 'random day agent' sends 2 notifications randomly in each day; (3) 'fixed agent' sends 2 notifications per day and they were evenly distributed (at 12:00 and 16:00). The performance of agents is shown in Fig. 3-left. We observed an obvious increase in the reward of ***R*** agent, while three others hold a relatively stable performance. It indicates our approach adaptively optimizes the policy to send a restricted amount of notifications with respect to user's momentary context, and afterward outperforms all context-blind agents.

Evaluation of restricted policy

In the second experiment, we evaluated the efficiency of our restricted notification setting and how well the RL agents perform when incorporating this constraint during the learning in two different ways. One is applied and described in our RL algorithm of Reinforcement Learning Algorithm (***R*** agent). Second is to integrate it into the simulation environment: after the maximum number of notifications is reached in an episode, a notification will not be sent even if the algorithm decides to send one (***C*** agent). In Fig. 3-right, we found that although the ***R*** agent learns faster than the ***C*** agent (consistent with results shown in [32]), two agents show a similar performance after learning. In addition, we set up the ***B*** agent, which had no restriction on the number of notifications sent in each episode before 15,000 episodes. Afterwards, we integrated the restriction in its environment, leading to a dramatic performance drop in Fig. 3-right. This phenomenon demonstrates the different performances from an agent without restriction during learning (agent ***B***) and agents with restriction during learning (both the agent ***R*** and agent ***C***). It indicates that the policy learned without considering the restriction hardly performs well in a restricted mHealth setting, suggesting the importance of modeling this practical restriction in training RL algorithms.

Interpretation of learned policy

We further evaluated our approach by visualizing the detailed information of episodes in the learning process. Results of episode No. 100, 1500 and 16000 in a run of the ***R*** agent are presented in Fig. 4, which correspond to a policy before learning, a policy at the end of the first rapid learning process and a policy at the stable stage of learning in Fig. 3-left. We observed that at the beginning stage (episode 100), the ***R*** agent sends all notifications early in the episode. Afterwards, the agent learns to spread the restricted number of notifications over the entire episode (see episode 1500). This is the first strategy our agent learns, which leads to the first increase of the reward in Fig. 3. Moreover, the ***R*** agent learns to send notifications based on contextual situations. Notifications are sent in the decision points with very bad situations (dark blue ones) in the first two episodes, but almost all of them are sent under very good situations (dark red ones) in episode 16000. Finally, as indicated in green color in Fig. 4, the ***R*** agent realized that the simulated users are unlikely to run again in the hours following a recent run. Hence, the strategy of 'not sending notification after a run' seems to be learned.

Conclusion and future work

In this paper, we explored the practical usage of adaptive and intelligent agents in personal mobile health intervention and developed an RL-based agent to optimize the strategy of adaptively delivering context-aware notifications. The simulation results showed that the policy learned by our RL agent is more efficient than manually defined strategies without context awareness. In particular, our work made two contributions to perform this practical learning task without bothering users too much. First, when incorporating prior knowledge from historical data and psychological theories for optimizing the policy, our proposed dynamic Bayes network can handle empirical data with various context space and flexible target activity. Second, we constrained notification frequency in a period and adapted an RL algorithm for this constraint. As far as we know, such constraint was never structurally studied and evaluated in a mHealth setting, our results provide evidence that it is essential to take the frequency restriction of certain actions into account in the learning process of RL. For future work, it would be interesting to examine the efficiency of various state-of-art RL algorithms considering this constraint. Also, the practical usage of our approach should be further evaluated in trials with real users. We have conducted a small-scale feasibility study [34]. Based on the initial results and learned lessons, we plan a longer study to evaluate the effectiveness of our pre-learned delivery strategy for comparable user groups.

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/s10916-021-01773-0.

Acknowledgements The authors thank the cooperator MYLAPS for providing the mobile application dataset, as well as anonymous reviewers for providing precious comments.

Funding This work is funded by Playful Data-driven Active Urban Living project under NWO and SIA grant 629.004.013.

Declarations

Research involving human and animal participants This article does not contain any studies with human participants or animals performed by any of the authors.

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

References

1. Collins, L.M., Murphy, S.A., Bierman, K.L.: A conceptual framework for adaptive preventive interventions. Prevention science **5**(3), 185–196 (2004)
2. Hardeman, W., Houghton, J., Lane, K., Jones, A., Naughton, F.: A systematic review of just-in-time adaptive interventions (Jitais) to promote physical activity. International Journal of Behavioral Nutrition and Physical Activity **16**(1), 31 (2019)
3. Nahum-Shani, I., Smith, S.N., Spring, B.J., Collins, L.M., Witkiewitz, K., Tewari, A., Murphy, S.A.: Just-in-time adaptive interventions (Jitais) in mobile health: key components and design principles for ongoing health behavior support. Annals of Behavioral Medicine **52**(6), 446–462 (2017)
4. Goldstein, S.P., Evans, B.C., Flack, D., Juarascio, A., Manasse, S., Zhang, F., Forman, E.M.: Return of the Jitai: applying a just-in-time adaptive intervention framework to the development of m-health solutions for addictive behaviors. International journal of behavioral medicine **24**(5), 673–682 (2017)
5. Sarker, H., Sharmin, M., Ali, A.A., Rahman, M.M., Bari, R., Hossain, S.M., Kumar, S.: Assessing the availability of users to engage in just-in-time intervention in the natural environment. In: Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing, pp. 909–920 (2014)
6. Thomas, J.G., Bond, D.S.: Behavioral response to a just-in-time adaptive intervention (Jitai) to reduce sedentary behavior in obese adults: Implications for Jitai optimization. Health Psychology **34**(S), 1261 (2015)
7. Yom-Tov, E., Feraru, G., Kozdoba, M., Mannor, S., Tennenholtz, M., Hochberg, I.: Encouraging physical activity in patients with diabetes: intervention using a reinforcement learning system. Journal of medical Internet research **19**(10), e338 (2017)
8. Gönül, S., Namlı, T., Coşar, A., and Toroslu, İ.H.: A reinforcement learning based algorithm for personalization of digital, just-in-time, adaptive interventions. Artificial Intelligence in Medicine **115**, 102062 (2021)
9. Aguilera, A., Figueroa, C.A., Hernandez-Ramos, R., Sarkar, U., Cemballi, A., Gomez-Pathak, L., Miramontes, J., Yom-Tov, E., Chakraborty, B., Yan, X., et al.: mhealth app using machine learning to increase physical activity in diabetes and depression: clinical trial protocol for the diamante study. BMJ open **10**(8), e034723 (2020)
10. Forman, E.M., Kerrigan, S.G., Butryn, M.L., Juarascio, A.S., Manasse, S.M., Ontañón, S., Dallal, D.H., Crochiere, R.J., Moskow, D.: Can the artificial intelligence technique of reinforcement learning use continuously-monitored digital data to optimize treatment for weight loss? Journal of behavioral medicine **42**(2), 276–290 (2019)
11. Rabbi, M., Aung, M.H., Zhang, M., Choudhury, T.: My behavior: automatic personalized health feedback from user behaviors and preferences using smartphones. In: Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing, pp. 707–718. ACM (2015)
12. Pellegrini, C.A., Pfammatter, A.F., Conroy, D.E., Spring, B.: Smartphone applications to support weight loss: current perspectives. Advanced health care technologies **1**, 13 (2015)
13. Saunders, W., Sastry, G., Stuhlmüller, A., Evans, O.: Trial without error: Towards safe reinforcement learning via human intervention. In: Proceedings of the 17th International Conference on

Autonomous Agents and MultiAgent Systems, pp. 2067–2069 (2018)
14. Zhou, M., Mintz, Y., Fukuoka, Y., Goldberg, K., Flowers, E., Kaminsky, P., Castillejo, A., Aswani, A.: Personalizing mobile fitness apps using reinforcement learning. In: CEUR workshop proceedings, vol. 2068. NIH Public Access (2018)
15. Tabatabaei, S.A., Hoogendoorn, M., van Halteren, A.: Narrowing reinforcement learning: Overcoming the cold start problem for personalized health interventions. In: International Conference on Principles and Practice of Multi-Agent Systems, pp. 312–327. Springer (2018)
16. Tomkins, S., Liao, P., Yeung, S., Klasnja, P., Murphy, S.: Intelligent pooling in thompson sampling for rapid personalization in mobile health (2019)
17. Gonul, S., Namli, T., Baskaya, M., Sinaci, A.A., Cosar, A., Toroslu, I.H.: Optimization of just-in-time adaptive interventions using reinforcement learning. In: International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems, pp. 334–341. Springer (2018)
18. Liao, P., Greenewald, K., Klasnja, P., Murphy, S.: Personalized heartsteps: A reinforcement learning algorithm for optimizing physical activity. Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies **4**(1), 1–22 (2020)
19. Ameko, M.K., Beltzer, M.L., Cai, L., Boukhechba, M., Teachman, B.A., Barnes, L.E.: Online contextual multi-armed bandits for mobile health interventions: A case study on emotion regulation. In: Fourteenth ACM Conference on Recommender Systems, pp. 249–258 (2020)
20. Sutton, R.S., Barto, A.G.: Reinforcement learning: An introduction. MIT press (2018)
21. Kamphorst, B., Kalis, A.: Why option generation matters for the design of autonomous e-coaching systems. AI & SOCIETY **30**(1), 77–88 (2015)
22. Tobias, R.: Changing behavior by memory aids: A social psychological model of prospective memory and habit development tested with dynamic field data. Psychological review **116**(2), 408–438 (2009)
23. Zhang, C., Lakens, D., IJsselsteijn, W.A.: Theory integration for lifestyle behavior change in the digital age: An adaptive decision-making framework. Journal of Medical Internet Research **23**(4), e17127 (2021)
24. Rubin, D.C., Hinton, S., Wenzel, A.: The precise time course of retention. Journal of Experimental Psychology: Learning, Memory, and Cognition **25**(5), 1161–1176 (1999)
25. Savage, L.J.: The foundations of statistics. Courier Corporation (1972)
26. Von Neumann, J., Morgenstern, O.: Theory of games and economic behavior. Bull. Amer. Math. Soc **51**(7), 498{504 (1945)
27. Dunton, G.F., Liao, Y., Intille, S., Huh, J., Leventhal, A.: Momentary assessment of contextual influences on affective response during physical activity. Health Psychology **34**(12), 1145 (2015)
28. Louro, M.J., Pieters, R., Zeelenberg, M.: Dynamics of multiple-goal pursuit. Journal of personality and social psychology **93**(2), 174 (2007)
29. Mihajlovic, V., Petkovic, M.: Dynamic bayesian networks: A state of the art. University of Twente Document Repository (2001)
30. Kaplan, W.: Advanced calculus. Pearson Education India (1952)
31. Williams, R.J.: Simple statistical gradient-following algorithms for connectionist reinforcement learning. Machine learning **8**(3–4), 229–256 (1992)
32. Fujita, Y., Maeda, S.i.: Clipped action policy gradient. In: Proceedings of the 35th International Conference on Machine Learning, *Proceedings of Machine Learning Research*, vol. 80, pp. 1597–1606. PMLR, Stockholmsmässan, Stockholm Sweden (2018)
33. Wang, S., Scheider, S., Sporrel, K., Deutekom, M., Timmer, J., Kröse, B.: What are good situations for running? a machine learning study using mobile and geographical data. Frontiers in Public Health **8**, 985 (2021)
34. Wang, S., Sporrel, K., van Hoof, H., Simons, M., de Boer, R.D., Ettema, D., Nibbeling, N., Deutekom, M. and Kröse, B.: Reinforcement learning to send reminders at right moments in smartphone exercise application: A feasibility study. International Journal of Environmental Research and Public Health, **18**(11), 6059 (2021)