Large-Scale Bandit Approaches for Recommender Systems

Qian Zhou[1], XiaoFang Zhang[1,2(✉)], Jin Xu[1], and Bin Liang[1]

[1] Department of Computer Science and Technology, Soochow University, Suzhou 215006, China
{20154227029,20154227016,20154227041}@stu.suda.edu.cn,
xfzhang@suda.edu.cn

[2] State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing 210033, China

Abstract. Recommender systems have been successfully applied to many application areas to predict users' preference. However, these systems face the exploration-exploitation dilemma when making a recommendation, since they need to exploit items which raise users' interest and explore new items to improve satisfaction simultaneously. In this paper, we deal with this dilemma through Multi-Armed Bandit (MAB) approaches, especially for large-scale recommender systems that have vast or infinite items. We propose two large-scale bandit approaches under the situations that there is no available priori information. The continuous exploration in our approaches can address the cold start problem in recommender systems. Furthermore, our context-free approaches are based on users' click behavior without the dependence on priori information. We theoretically prove that our approaches can converge to optimal item recommendations in the long run. Experimental results indicate that our approaches are able to provide more accurate recommendations than some classic bandit approaches in terms of click-through rates, with less calculation time.

Keywords: Recommender systems · Multi-Armed Bandit · Context-free

1 Introduction

The primary target of recommender systems is to propose one or several items to users in which they might be interested. The books, articles or music provided by the recommender systems are items [1, 2]. Recommender systems need to focus on items that raise users' interest and explore new items to improve users' satisfaction at the same time. That creates an exploration-exploitation dilemma, which is the core point of Multi-Armed Bandit (MAB) problems [3]. Exploration means learning new items' payoff for a particular user by recommending new items. Exploitation means recommending the best items based on the payoffs observed so far. The payoff of a recommendation is widely measured by Click-Though Rate (CTR) [4]. Then the goal of recommendations is to maximize the CTR over all users.

Contextual bandit approaches are already studied in many fields of recommender systems [5]. In large-scale recommender systems, there are large or infinite number of

D. Liu et al. (Eds.): ICONIP 2017, Part I, LNCS 10634, pp. 811–821, 2017.
https://doi.org/10.1007/978-3-319-70087-8_83

contexts, as a sequence, the increasing recommenders based on contexts fail to ensure effective and efficient recommendations. There are several context-aware bandit approaches can be applied in large-scale recommender systems [6, 7].

However, there exist some recommender systems that the priori information about the items and users is unknown, the cold start problem appears when the system has no priori information in practice [8, 9]. Under this situation, recommendation has to be inferred from user feedbacks. As a result, contrary to the contextual case, our work focuses on context-free case. Some of the existing context-free bandit approaches fail to make full use of user feedback or do not apply to large-scale problems. We attempt to design cost-effective approaches without dependence on priori information for large-scale recommender systems. Each item corresponds to an action (referred to as the arm in a bandit framework) in our work.

We propose two context-free bandit approaches which try to address all of the above mentioned challenges in large-scale recommender systems. The recommendation is made only based on the payoff estimations without dependence on any priori information. The cold start issue is addressed by continuously exploration. Our approaches are proved to converge to optimal item recommendations in the long run. Experiments are made on Yahoo! Front Page Today Module user click log dataset. Our approaches are able to achieve higher CTRs than some existing bandit approaches, such as EXP3 and UCB1, with less calculation time.

The rest of the paper is organized as follows. Section 2 presents some related works. In Sect. 3, we introduce our approaches, discuss the influence of key parameters and prove the convergence. Section 4 presents experimental evaluation. Conclusion is made in Sect. 5.

2 Related Work

Recommender systems have been successfully applied to many application areas to predict users' preference. Two main categories of recommendation algorithms are filtering-based and reinforcement learning methods [8]. In this paper, we focus on reinforcement learning methods. Reinforcement learning methods, such as MAB and Markov Decision Processes (MDPs) [10], are widely used in recommender systems. MDP-based approaches model the last k choices of a user as the state and the available items as the action set to maximize the long-run payoff. However, MDP-based approaches suffer very slow convergence rates in large-scale recommender systems [11].

MAB-based approaches make recommendations by balancing between exploration and exploitation, such as ε-greedy [12], softmax [13], EXP3 [14] and UCB1 [3]. Among these context-free approaches, ε-greedy is the simplest approach, but the performance of ε-greedy is still always competitive. Softmax makes recommendations according to a probability distribution based on user feedbacks. As a complicated variant of softmax, the main idea of EXP3 is to divide the payoff of an item by its chosen probability. UCB1 always recommends the item with the highest upper confidence index. However, UCB1 needs to sweep all items during the initial period, which may be inappropriate for large-scale recommender systems. Contexts are considered, aiming at improving the

effectiveness of recommendations further. Generally, contexts represent the situations of the user when a recommendation is made, such as time, gender, and search query [15, 16]. The LinUCB algorithm is proposed to solve news article recommendation problems [17]. The Naive III and Linear Bayes approaches define a user-group by a set of features that individual users may have in common [7]. A MAB-based clustering approach constructs an item-cluster tree for recommender systems [6].

3 Our Approaches

In this section, we present two context-free bandit approaches for large-scale problems. The first approach is based on the Chosen Number of Action with Minimal Estimation, namely CNAME. Then we introduce an asynchronous CNAME approach, namely Asy-CNAME.

3.1 CNAME Approach

Some of the existing context-free bandit approaches fail to make full use of user feedback, such as ε-greedy, or do not apply to large-scale problems, such as UCB1. Therefore, the CNAME approach is proposed to address these two issues. The key idea of CNAME is how to use user feedbacks sufficiently. Both the estimated payoff and the chosen number of an action are utilized to update exploration probability. The CNAME approach is presented in Algorithm 1.

Algorithm 1.CNAME

Input: $w > 0$
for *each action* k in possible action set **do**
  $Q(k) \leftarrow 0$
  $N(k) \leftarrow 0$
end
for *time step* $t \leftarrow 1$ to T **do**
  $m_t \leftarrow N\left(\arg\min_k Q(k)\right)$
  $p \leftarrow \frac{w}{w + m_t^2}$
  Generate a random number x in open interval (0,1)
  $a_t \leftarrow \begin{cases} \arg\max_k Q(k) & \text{if } x > p \\ \text{a random action} & \text{otherwise} \end{cases}$
  Observe a reward $X_{a_t,t}$
  $N(a_t) \leftarrow N(a_t) + 1$
  $Q(a_t) \leftarrow Q(a_t) + \frac{1}{N(a_t)}[X_{a_t,t} - Q(a_t)]$
end

The CNAME starts by setting the parameter w (Line 1). The parameter w affects the speed at which the exploration probability is changed. After initializing the estimation and the chosen number of each action k (Line 3–4), it iteratively chooses an action to play (referred to recommend an item in recommender systems) based on the exploration probability (Line 7–10). Finally, the CNAME updates the number of chosen and estimation at time step t (Line 12–13). The exploration probability p is adjusted according to the chosen number of action with minimal estimated payoff, defined by m_t.

The CNAME approach has three points. Firstly, the influence of m_t on exploration increases with decreasing w, and vice versa. Thus, the parameter w can change the effect of user feedbacks on exploration probability. Secondly, the increasing of m_t means action with the lowest estimated payoff is chosen. Such action can be the least contribution to the entire payoff. As m_t increases, the exploration probability will be reduced. That means the chosen probability of greedy action (action with the highest estimation) will be increased, which can help to improve the actual gain of entire payoff. Thirdly, the CNAME algorithm explores continuously to help to learn the payoffs of new items.

3.2 Asynchronous CNAME Approach

Aiming at ensuring the effective and efficient recommendations for large-scale recommender systems, the CNAME approach should be updated in an asynchronous manner.

Algorithm 2. Asy-CNAME

1. **Input**: $w>0$, $p \leftarrow 0.9$ and $\alpha \in (0,1)$
2. **for** *each action* k in possible action set **do**
3. $\quad Q(k) \leftarrow 0$
4. $\quad N(k) \leftarrow 0$
5. **end**
6. **for** $i \leftarrow 1$ to M **do**
7. $\quad$ **for** $j \leftarrow 1$ to N **do**
8. $\quad\quad t \leftarrow Mi + j$
9. $\quad\quad$ Generate a random number x in open interval (0,1)
10. $\quad\quad a_t \leftarrow \begin{cases} \arg\max_k Q(k) & \text{if } x > p \\ \text{a random action} & \text{otherwise} \end{cases}$
11. $\quad\quad$ Observe a reward $X_{a_t,t}$
12. $\quad\quad N(a_t) \leftarrow N(a_t) + 1$
13. $\quad\quad Q(a_t) \leftarrow Q(a_t) + \frac{1}{N(a_t)}[X_{a_t,t} - Q(a_t)]$
14. $\quad$ **end**
15. $\quad m_t \leftarrow N\left(\arg\min_k Q(k)\right)$
16. $\quad p' \leftarrow \frac{w}{w + m_t^2}$
17. $\quad$ Update p as $p \leftarrow (1-\alpha)p + \alpha p'$
18. **end**

The Asy-CNAME approach is presented in Algorithm 2. Different from the CNAME, the Asy-CNAME clusters a sequence of N samples of the action into a single batch (Line 8–13) and updates the exploration probability after each batch ends (Line 15–17), where the terminal time step $T = MN$ (Line 6–7). Note that at the end of each batch, the estimated expected payoff of some of the actions may not have improved at all. Therefore, a smoothing mechanism is needed (Line 17), to avoid being overcommitted to the new estimate of different actions.

For the CNAME approach, the exploration probability is updated after an action is chosen each time. This may lead to a recommendation that is too susceptible to user's recent behavior. Thus the Asy-CNAME approach updates exploration probability in batches. Asynchronous manner weakens the impact of the user's short-term behavior to a certain extent, which plays a role in improving the CTR. On the other hand, the Asy-CNAME approach reduces the implementation complexity by asynchronous manner, which can help to decrease the calculation time.

3.3 Convergence of Our Approaches

Based on the above description of our approaches, we prove that proposed approaches are able to converge to the optimum in the long run.

A K-armed bandit problem is defined by random variables $X_{i,n}$ for $1 \leq i \leq K$ and $n \geq 1$. Each i represents an action (referred to the arm of a bandit) and K is the number of actions and n refers to the number of trials. Successive trials of action i yield rewards $X_{i,1}$, $X_{i,2}$…which are independent and identically distributed according to an unknown law with unknown expectation μ_i. Note that given $\mu_1, \ldots, \mu_K$, we define the action i with $\mu_i = \mu^*$ as an optimal action. In what follows, we write $\overline{X}_n^*$ and N_n^* instead of $\overline{X}_{i,n}$ and $N_n(i)$, where i is the optimal action. Here

$$\overline{X}_{i,n} = \frac{1}{n} \sum_{t=1}^{n} X_{i,t}$$

The CNAME and Asy-CNAME are algorithms that choose the next action based on the sequence of past trials and obtained payoffs. Let $N_n(i)$ be the number of times action i has been chosen by the CNAME and Asy-CNAME during the first n trials. Of course, we always have

$$\sum\nolimits_{i=1}^{K} N_n(i) = n$$

Let

$$\varepsilon_t = \frac{w}{w + m_t^2}, x_0 = \frac{1}{2K} \sum_{t=1}^{n} \varepsilon_t \text{ and } n > \frac{2}{w}$$

The probability that action i is chosen at trial n is

$$P\{a_n = i\} \leq \frac{\varepsilon_n}{K} + \left(1 - \frac{\varepsilon_n}{K}\right) P\left\{\overline{X}_{i,N_{n-1}(i)} \geq \overline{X}^*_{N^*_{n-1}}\right\} \tag{1}$$

and

$$P\left\{\overline{X}_{i,N_n(i)} \geq \overline{X}^*_{N^*_n}\right\} \leq P\left\{\overline{X}_{i,N_n(i)} \geq \mu_i + \frac{\Delta_i}{2}\right\} + P\left\{\overline{X}^*_{N^*_n} \leq \mu^* - \frac{\Delta_i}{2}\right\} \tag{2}$$

Where $\Delta_i = \mu^* - \mu_i$. Then we have

$$\begin{aligned} P\left\{\overline{X}_{i,N_n(i)} \geq \mu_i + \frac{\Delta_i}{2}\right\} &= \sum_{t=1}^{n} P\left\{N_n(i) = t \wedge \overline{X}_{i,t} \geq \mu_i + \frac{\Delta_i}{2}\right\} \\ &= \sum_{t=1}^{n} P\left\{N_n(i) = t \mid \overline{X}_{i,t} \geq \mu_i + \frac{\Delta_i}{2}\right\} \cdot P\left\{\overline{X}_{i,t} \geq \mu_i + \frac{\Delta_i}{2}\right\} \end{aligned}$$

Let $N^R_n(i)$ be the number of plays in which action i was chosen at random in the first n trials. By using the Chernoff-Hoeffding bound, we get

$$\begin{aligned} \sum_{t=1}^{n} P\left\{N_n(i) = t \,\middle|\, \overline{X}_{i,t} \geq \mu_i + \frac{\Delta_i}{2}\right\} \cdot P\left\{\overline{X}_{i,t} \geq \mu_i + \frac{\Delta_i}{2}\right\} &\leq \sum_{t=1}^{n} P\left\{N_n(i) = t \,\middle|\, \overline{X}_{i,t} \geq \mu_i + \frac{\Delta_i}{2}\right\} \cdot e^{-\frac{\Delta_i^2 t}{2}} \\ &\leq \sum_{t=1}^{\lfloor x_0 \rfloor} P\left\{N^R_n(i) \leq t \,\middle|\, \overline{X}_{i,t} \geq \mu_i + \frac{\Delta_i}{2}\right\} + \frac{2}{\Delta_i^2} e^{-\frac{\Delta_i^2 \lfloor x_0 \rfloor}{2}} \\ &\leq x_0 \cdot P\left\{N^R_n(i) \leq x_0\right\} + \frac{2}{\Delta_i^2} e^{-\frac{\Delta_i^2 \lfloor x_0 \rfloor}{2}} \end{aligned} \tag{3}$$

In the last line we dropped the conditioning because each action is chosen at random independently of the previous choices of the algorithm. Since

$$E[N^R_n(i)] = \frac{1}{K}\sum_{t=1}^{n} \varepsilon_t \text{ and } \operatorname{Var}\left[N^R_n(i)\right] = \sum_{t=1}^{n} \frac{\varepsilon_t}{K}\left(1 - \frac{\varepsilon_t}{K}\right) \leq \frac{1}{K}\sum_{t=1}^{n} \varepsilon_t$$

by the Bernstein's inequality we get

$$P\left\{N^R_n(i) \leq x_0\right\} \leq e^{-\frac{x_0}{5}} \tag{4}$$

Finally it remains to lower bound x_0

$$\begin{aligned} x_0 &= \frac{1}{2K}\sum_{t=1}^{n}\frac{w}{w+m_t^2} \\ &\geq \frac{1}{2K}\sum_{t=1}^{n}\frac{w}{w+t^2} \\ &\geq \frac{wn-2}{2Kw} \end{aligned} \tag{5}$$

Then, using (1)–(4) and the above lower bound on x_0 we obtain

$$\begin{aligned} P\{a_n = j\} &\leq \frac{\varepsilon_n}{K} + 2\left(x_0 e^{-\frac{x_0}{5}} + \frac{2}{\Delta_i^2}e^{-\frac{\Delta_i^2\lfloor x_0\rfloor}{2}}\right) \\ &\leq \frac{w}{\left(w+m_n^2\right)K} + \frac{wn-2}{Kw}e^{-\frac{wn-2}{10Kw}} + \frac{4}{\Delta_i^2}e^{-\frac{\Delta_i^2(wn-2)}{4Kw}} \\ &\leq \frac{1}{K} + \frac{wn-2}{Kw}e^{-\frac{wn-2}{10Kw}} + \frac{4}{\Delta_i^2}e^{-\frac{\Delta_i^2(wn-2)}{4Kw}} \end{aligned} \tag{6}$$

For all $K \geq 1$ and for all reward distributions with support in [0, 1], the probability that the CNAME and Asy-CNAME algorithms choose a suboptimal action i is at most

$$\frac{1}{K} + \frac{wn-2}{Kw}e^{-\frac{wn-2}{10Kw}} + \frac{4}{\Delta_i^2}e^{-\frac{\Delta_i^2(wn-2)}{4Kw}}$$

For $n \to \infty$ and K large enough the above bound is 0. It means the CNAME and Asy-CNAME algorithms are able to converge to the optimal action in large-scale MAB problems. This concludes the proof.

4 Experimental Evaluation

In this section, we discuss the influence of key parameter w, learning rate α and different update manners in our approaches. We provide the reference ranges of parameter w and learning rate α through simulation on a randomly generated dataset. Then we compare the performance of our approaches with other bandit approaches on Yahoo! Front Page Today Module user click log dataset.

4.1 Randomly Generated Dataset

The goal of this simulation is to minimize the regret [18], which is the loss between the optimal expected total payoff and the expected total payoff gained through our approaches. Eventually, the smaller value of regret implies the better performance.

The subject is a set of tasks with 100 randomly generated K-armed bandit problems. The actual value of each task $\boldsymbol{\mu} = [\mu_1, \ldots, \mu_K]$ is a Gaussian distribution with a mean of 0 and a variance of 1. The reward for each action i is subject to a Gaussian distribution with a mean of μ_i and a variance of 1.

Experimental Evaluation about Key Parameters. Under the above experimental conditions, we take different values of the parameter w and learning rate α, and record the average regrets of 100 random tasks, where the number of actions $K = 1000$, batch $N = 10$ and terminal time step $T = 2000$.

In Fig. 1, with the increasing values of w, the difference between the CNAME and Asy-CNAME approaches on the average regret is reduced. When parameter w is in the interval [0.01, 0.1], the average regret is relatively low, as shown in Fig. 1. Thus, we use interval [0.01, 0.1] as the reference range of w. Figure 2 shows that the average regret of the Asy-CNAME approach is lower than the CNAME approach when $\alpha > 0.3$. The Asy-CNAME approach performs best when $\alpha = 0.6$.

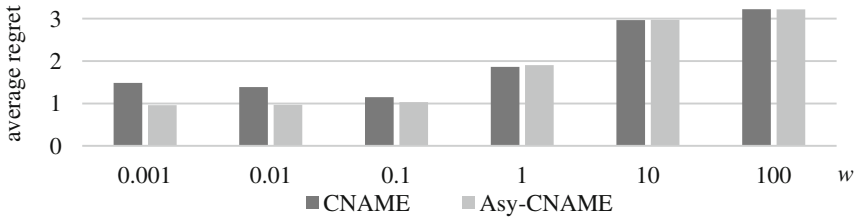


Fig. 1. The average regret obtained by the CNAME algorithm and the Asy-CNAME algorithm respectively with different values of parameter w when $\alpha = 0.8$

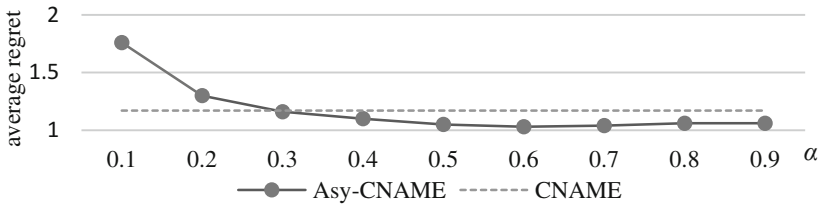
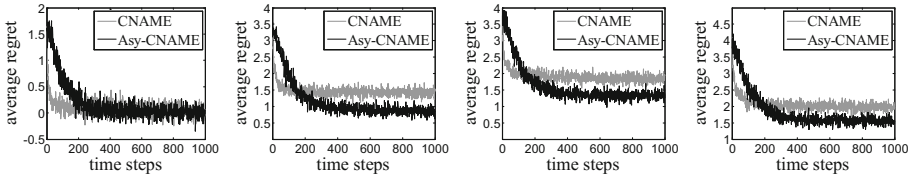


Fig. 2. The average regret obtained by the CNAME algorithm and Asy-CNAME algorithm with different values of learning rate α when $w = 0.1$

Experimental Evaluation about Different Update Manners. The purpose of this part is to compare influence of different update manners in large-scale MAB problems. We

compare the CNAME and Asy-CNAME with $K = 100$, 1000, 5000, 10000 respectively, where $\alpha = 0.8$, $w = 0.1$, $N = 10$ and $T = 1000$. The experiment results are presented in Fig. 3.



(a) K=100 (b) K=1000 (c) K=5000 (d) K=10000

Fig. 3. The average regret obtained by the CNAME algorithm and the Asy-CNAME algorithm respectively with different values of K

In Fig. 3, the average regrets of both approaches increase with the increasing values of K. Figure 3 shows that the convergence rate of CNAME is faster than Asy-CNAME and the average regret of CNAME is lower than Asy-CNAME at the beginning. The final average regrets obtained by Asy-CNAME are lower than those of CNAME when K is large enough ($K > 100$). Since the exploration probability of CNAME is updated synchronously, the CNAME prefers greedy action at the beginning. The Asy-CNAME spends more time learning new actions by exploration since it updates asynchronously. Learning news actions can help to improve payoff in the long run.

4.2 Yahoo! Front Page Today Module User Click Log Dataset

This dataset contains a fraction of user click log for news articles displayed in the Featured Tab of the Today Module on Yahoo! Front Page[1]. This dataset includes 15 days of data from October 2 to 16, 2011 and raw features. There are 28,041,015 user visits to the Today Module on Yahoo!'s Front Page.

In this part, we make recommendations for large-scale recommender systems, through MAB-based approaches. The Random approach randomly chooses an item each time. This can be seen as the benchmark for other approaches. Although we focus on the context-free situations, our approaches can be applied to context-aware situations directly. So in addition to context-free approaches, we compare our approaches with a context-aware approach named Naive III. The performances of approaches are evaluated through CTR as shown in Table 1. In Table 1, the best results are highlighted respectively in boldface.

[1] https://webscope.sandbox.yahoo.com.

Table 1. Performance comparison in CTR and calculation time on the Yahoo! Front Page Today Module user click log dataset.

Algorithm	Lines					Time (min) on 1.4×10^7 lines
	2.0×10^5	3.6×10^6	7.2×10^6	1.06×10^7	1.4×10^7	
Random	0.036	0.034	0.034	0.034	0.034	**15.017**
ε-greedy	0.046	0.065	0.065	0.066	0.067	21.167
Softmax	0.040	0.041	0.041	0.041	0.041	23.533
UCB1	0.037	0.049	0.052	0.055	0.056	27.133
EXP3	0.039	0.040	0.040	0.040	0.040	23.250
Naive III	**0.047**	0.066	0.067	0.068	0.069	50.267
CNAME	0.043	0.067	**0.069**	**0.070**	0.071	23.033
Asy-CNAME	0.044	**0.068**	**0.069**	**0.070**	**0.072**	21.017

In terms of CTR, it can be calculated from the data in Table 1 that the CNAME approach achieves a 6%–109% performance gain over other context-free approaches, including the Random, ε-greedy, softmax, EXP3 and UCB1. Over the first 200,000 rows, the Naive III yields the highest CTR. After that, the CTRs of CNAME and Asy-CNAME are even higher than those of Naive III approach. In the comparison of time, the Random approach consumes the least calculation time as the benchmark. The Asy-CNAME takes the second least calculation time. The CNAME approach consumes similar time with other context-free approaches, such as softmax and EXP3, while obtaining higher CTR. The context-free Asy-CNAME approach just needs about 21 min to obtain the CTR over the first 14,000,000 rows on Yahoo! dataset, while the context-aware Naive III approach consumes about 50 min to get CTR over the same rows. On the other side, as context-free approaches, the CNAME and Asy-CNAME can be applied to different recommender systems easily. In a summary, the CNAME and Asy-CNAME approaches achieve higher CTR with comparable calculation time. Thus, the CNAME and Asy-CNAME approaches are cost-effective for large-scale recommender systems.

5 Conclusion

In this paper, we study recommender systems based on large-scale MAB problems. The CNAME and the Asy-CNAME approaches make good recommendations without dependence on priori information. The cold start problem is addressed by continuous exploration in our approaches.

Theoretical result shows that our approaches are able to converge to the optimal recommendations in the long run. The reference range of key parameters are given through our simulation. Besides, the performance of our approaches and other MAB-based recommendation approaches is compared on Yahoo! Front Page Today Module user click log dataset. Experimental results show that our approaches outperform other algorithms in terms of CTR. The CNAME and Asy-CNAME approaches are cost-effective for large-scale recommender systems. Although our approaches achieve significant result, a possible improvement can be made by using contexts rationally if there are available priori information.

References

1. Resnick, P., Varian, H.R.: Recommender systems. Commun. ACM **40**(3), 56–58 (1997)
2. Balabanović, M., Shoham, Y.: Fab: content-based, collaborative recommendation. Commun. ACM **40**(3), 66–72 (1997)
3. Auer, P., Cesa-Bianchi, N., Fischer, P.: Finite-time analysis of the multiarmed bandit problem. Mach. Learn. **47**(2), 235–256 (2002)
4. Liu, J., Dolan, P., Pedersen, E. R.: Personalized news recommendation based on click behavior. In: International Conference on Intelligent User Interfaces, pp. 31–40 (2010)
5. Tang, L., Jiang, Y., Li, L., Li, T.: Ensemble contextual bandits for personalized recommendation. In: RecSys, pp. 73–80 (2014)
6. Song, L., Tekin, C., Schaar, M.V.D.: Online learning in large-scale contextual recommender systems. IEEE Trans. Serv. Comput. **9**(3), 433–445 (2016)
7. Jośe, A.M.H., Vargas, A.M.: Linear Bayes policy for learning in contextual-bandits. Expert Syst. Appl. **40**(18), 7400–7406 (2013)
8. Adomavicius, G., Tuzhilin, A.: Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. IEEE Trans. Knowl. Data Eng. **17**(6), 734–749 (2005)
9. Su, X., Khoshgoftaar, T.M.: A survey of collaborative filtering techniques. Adv. Artif. Intell. **2**(1), 1–19 (2009)
10. Shani, G., Heckerman, D., Brafman, R.I.: An MDP-based recommender system. J. Mach. Learn. Res. **6**(1), 1265–1295 (2005)
11. Ren, Z., Krogh, B.H.: State aggregation in markov decision processes. In: IEEE Conference on Decision and Control, pp. 3819–3824 (2002)
12. Cesa-Bianchi, N., Lugosi, G.: Prediction, Learning, and Games. Cambridge University Press, Cambridge (2006)
13. Cesa-Bianchi, N., Fischer, P.: Finite-time regret bounds for the multi-armed bandit problem. In: ICML, pp. 100–108 (1998)
14. Bubeck, S., Slivkins, A.: The best of both worlds: stochastic and adversarial bandits. J. Mach. Learn. Res. **23**(42), 1–23 (2012)
15. Adomavicius, G., Tuzhilin, A.: Context-aware recommender systems. In: Ricci, F., Rokach, L., Shapira, B. (eds.) Recommender Systems Handbook, pp. 191–226. Springer, Boston (2015). doi:10.1007/978-1-4899-7637-6_6
16. Adomavicius, G., Sankaranarayanan, R., Sen, S., Tuzhilin, A.: Incorporating contextual information in recommender systems using a multidimensional approach. ACM Trans. Inf. Syst. **23**(1), 103–145 (2005)
17. Li, L., Chu, W., Langford, J., Schapire, R.E.: A contextual-bandit approach to personalized news article recommendation. In: World Wide Web, pp. 661–670 (2010)
18. Bubeck, S., Cesa-bianchi, N.: Regret analysis of stochastic and nonstochastic multi-armed bandit problems. Found. Trends Mach. Learn. **5**(1), 1–122 (2012)