Multi-Armed Bandit Algorithms and Their Applications in Online Recommendation Systems

Zijie Zhao

{zhaozijie_2024@sina.com}

Polytechnic Institute, Purdue University, Lafayette, 47901, United States of America

Abstract: Recommendation systems are used in various online services like streaming media services, e-commerce sites and news apps to enable users to find content of their choice quickly. This often involves sequential decision-making without knowing the rewards or pay-offs that the recommendations yield. A classic example of such decision-making is the multi-armed bandit (MAB) framework. This paper gives an overview of the basic MAB algorithms and their use cases for online recommendation systems. The paper covered some of the popular bandit algorithms, such as Upper Confidence Bound (UCB), Thompson Sampling, contextual bandits, off-policy evaluation, etc. Bandit algorithms are instrumental in solving the exploration-exploitation dilemma in the recommendation problem. They can learn the user preference and meanwhile, keep optimizing the quality of the item list. However, there are still a few practical problems, such as noisy feedback, large action space, and non-stationarity of user behavior, that remain to be solved.

Keywords: Multi-armed bandit, recommendation systems, exploration–exploitation

1 Introduction

Considering the proliferation of online platforms such as video streaming services, online shopping websites, and news apps, recommendation systems have become an integral part of assisting users in accessing relevant information in an efficient manner. Recommendation systems are required to make decisions under uncertain conditions; therefore, models such as the multi-armed bandit (MAB) are used naturally. This problem has a rich history, and it is the simplest model that incorporates both exploration and exploitation. There are many online systems, for e.g., video recommendation, e-commerce recommendation, news recommendation, etc. In these systems, at each time step, a decision is made as to which item to show the user, and the feedback is noisy, i.e., the same user may click on the item on one day, but may not click on the same item on another day. Further, in most of these systems, there is an additional constraint of having limited time and limited traffic, and trying all options roughly equally may imply foregoing rewards.

This paper aims to give an overview of the popular MAB algorithms and their use in online recommendation systems. Several studies have explored the development of multi-armed

ICIAAI 2026, May 22-24, Singapore, Singapore

DOI 10.4108/eai.22-5-2026.2365249

bandit algorithms and their applications in recommendation systems. Auer et al. analyzed the UCB algorithm and provided theoretical bounds on its performance in stochastic bandit settings[1]. Agrawal and Goyal studied Thompson Sampling and showed that it achieves near-optimal regret in many practical scenarios[2]. Li et al. applied contextual bandit algorithms to personalized news recommendation, demonstrating their effectiveness in real-world online systems[3]. In addition, Dudík et al. proposed methods for off-policy evaluation, which allow researchers to estimate the performance of recommendation policies using historical data[4]. In Section 2, this paper presents the necessary definitions, terminology, and classification of MAB algorithms. Section 3 describes the major MAB algorithms and use cases in online recommendation systems in section 3, and briefly discusses the extension to online ad serving, which is a new online decision-making problem. Section 4 discusses challenges and future directions, and Section 5 presents the conclusion.

2 Fundamentals of the Multi-Armed Bandit Problem

Thus, in these systems, it is necessary to carefully balance exploration and exploitation. In the MAB problem, there are a few slot machines with arms. At each step, a slot machine is chosen, and its arm is pulled, and it returns some reward, which could be binary, indicating success or failure, or real valued. The goal is to make a sequence of choices of slot machines and their pulls, to maximize the cumulative reward. Clearly, there is an exploration-exploitation tradeoff here: if it were known which slot machine had the largest expected reward, then the arm of that machine could be repeatedly pulled to maximize the cumulative reward. However, this information is not known beforehand, and exploration is required to identify which slot machine has the maximum expected reward. The MAB problem is simple enough to formalize this exploration-exploitation tradeoff and develop algorithms to decide which slot machine to pull at each step[5].

There are many bandit algorithms out there. In the classical stochastic setting, one of the most popular ones is the UCB algorithm, which chooses actions based on the rewards they have received so far as well as on the uncertainty of these estimates. The strengths of UCB are that it provides a concrete and theoretically justified way of trading-off exploration and exploitation, and that there are concrete finite time regret bounds for this class of algorithms.

Another very popular algorithm is Thompson sampling, where actions are chosen by sampling from the posterior distribution over the rewards. The advantages of this algorithm are that it is often very easy to implement in practice and that it often works very well empirically. The caveat is that the performance of the algorithm depends on the choice of distribution over rewards.

More generally, bandit algorithms vary in their computational requirements, in their assumptions over the reward distribution, or in whether they can easily incorporate side information. These considerations are particularly relevant in the context of recommendation systems, where typically features of the user and the items are available. Indeed, bandit variants such as contextual bandits have become very important for online personalization and recommendation systems.

3 Applications of Multi-Armed Bandit Algorithms in Online Recommendation

Multi-armed bandit methods offer a useful framework for recommendation under uncertainty and have been extensively studied in recommender systems [6 – 9].

In the standard stochastic bandit setting, UCB selects actions by combining estimated rewards with uncertainty terms [1], while Thompson Sampling selects actions based on samples drawn from posterior reward distributions [2].

For personalized recommendation, contextual bandits can incorporate information about users and items [3, 7, 8, 10].

When online experimentation is costly or risky, offline evaluation techniques, such as the doubly robust estimator, can be used to evaluate new policies using historical data [4].

A good example of a scenario that can be solved with the help of a multi-armed bandit algorithm is a system that needs to make personalized news recommendations to its users. The system needs to select which news articles to display to its users while having limited information about their preferences. Contextual bandit algorithms can be used for this task, allowing the system to learn user preferences over time. This can help the system make more accurate recommendations while at the same time exploring new content that the user may enjoy.

Another common use for multi-armed bandit algorithms is in online advertising. Online advertisers need to select which advertisements to display to their users in order to get the highest level of user engagement. A system can use a bandit algorithm for this task, allowing it to experiment with different advertisements while gradually focusing on the ones that perform better. This can be more cost-effective compared to traditional A/B testing.

4 Challenges and Future Directions

While bandit algorithms are appealing for real-time decision making, there are many challenges when implementing these algorithms in a real-world recommender system. For one, the reward signal can be noisy and non-stationary. For example, a user may click on an item today, but not click on it tomorrow. Similarly, the reward signal can change throughout the day, week, month, or year based on user preferences and seasonality. This makes it hard to assume a fixed reward distribution, which is commonly assumed in the classic stochastic bandit setting.

Another challenge is that many real-world recommender systems operate at scale. For example, there may be millions of candidate arms (items) to choose from, which can be costly (e.g. , in terms of traffic loss and user experience) to explore. Cold-start remains a challenge. For example, new users or items have limited data, which can lead to slow convergence or high variance in the beginning. Evaluation is also challenging since online A/B testing comes with the risk of losing users and traffic. Although analysis can be performed on logged data, it is often biased by previous policies, and therefore, evaluating the performance of a new policy requires off-policy evaluation techniques. Lastly, real-world systems may have additional

constraints such as fairness, safety, and business rules. Pure exploration can incur regret in the short-term and may deteriorate the user experience if bad items are displayed too frequently.

These difficulties demonstrate that bandit-based recommendation systems in real applications involve not only developing advanced algorithms, but also system design and performance evaluation. One potential approach to tackle non-stationary reward signals in bandit-based recommendation is to develop algorithms that can quickly adapt their parameter estimates when the reward distribution changes. In particular, algorithms in this family typically use sliding window or change-point detection approaches to adapt to changing user preferences over time. In real applications where the number of candidate items is massive (e.g., millions of items), scalable bandit algorithms and efficient sampling schemes are essential. For instance, bandit algorithms that integrate with dimensionality reduction or candidate filtering could be potentially used to reduce the exploration cost while maintaining the quality of the recommended items. To further evaluate the performance of bandit algorithms, more robust and reliable off-policy evaluation methods are needed to estimate the performance of learned policies from logged interaction data. For example, doubly robust estimators and counterfactual learning have been applied to better estimate the performance of the policies and mitigating the bias in evaluation. Future work may be directed towards developing adaptive bandit algorithms in bandit-based recommendation, which can be easily implemented to adapt to the dynamically changing environment, scale up to a large number of items, and incorporate richer contextual information for personalizing the recommendation.

5 Conclusion

The paper introduced the MAB model, and why they are a good fit for online recommendation. Bandits are a natural fit because platforms must act sequentially and only receive feedback in the form of user behavior when an item is shown, and this feedback itself is noisy; the same user may click on the same item one day and ignore it the next. A model that can learn from feedback while making reasonable decisions is thus necessary.

A common concept underlying many of the bandit algorithms introduced in this paper is the need to balance exploration and exploitation. Intuitively, a bandit algorithm that purely exploits will potentially converge to a local maximum and will not have the opportunity to learn the reward distributions of other actions, while a bandit algorithm that purely explores will waste valuable traffic and compromise the user experience. UCB-style algorithms circumvent this issue by adding an “optimism bonus” to actions that have been explored less, incentivizing the algorithm to continue to explore actions with less certain reward distributions. Thompson Sampling accomplishes the same objective with a sampling-based algorithm, which can be more intuitive in practice. The paper also introduces algorithms for contextual bandits, as a single strategy is rarely sufficient for all users. For example, the behavior history of the user, the category of the item, or the time of day may all impact the reward distribution of a recommendation. Furthermore, the paper introduces algorithms for off-policy evaluation, which is often necessary in scenarios where online A/B testing is prohibitively costly or risky. In many applications, it is difficult or impossible to run experiments on live traffic, and therefore, off-policy evaluation is a convenient tool for approximating the performance of a given policy with logged data.

Bandits however do not solve the full recommendation problem on their own. Real-world recommendation systems often involve very large item sets, fast evolving user preferences, and often have constraints such as satisfying a user safety or satisfaction criteria. Going forward, research may focus on bandit algorithms that incorporate change-point detection to react more quickly to changing user preferences, deal with huge action spaces better, and be able to more effectively evaluate candidate policies before rolling out. Bandits still provide a good starting point however, since they offer a mathematical framework to think about learning from interactions, and online recommendation systems deal with interactions all day long.

6 Declaration of AI Use

During the preparation of this manuscript, the author used ChatGPT (OpenAI) only for language editing and proofreading purposes, such as improving grammar, clarity, and readability. The content, analysis, interpretation, and conclusions of the manuscript were reviewed and written by the author. All references and citation information were independently checked and verified by the author. The author has reviewed and approved the final version of the manuscript and takes full responsibility for its content.

References

[1] P. Auer, N. Cesa-Bianchi, and P. Fischer, “Finite-time analysis of the multiarmed bandit problem,” Machine Learning, vol. 47, nos. 2–3, pp. 235–256, 2002, https://doi.org/10.1023/A:1013689704352.

[2] S. Agrawal and N. Goyal, “Further optimal regret bounds for Thompson sampling,” in Proceedings of the Sixteenth International Conference on Artificial Intelligence and Statistics, PMLR, vol. 31, pp. 99–107, 2013.

[3] L. Li, W. Chu, J. Langford, and R. E. Schapire, “A contextual-bandit approach to personalized news article recommendation,” in Proceedings of the 19th International Conference on World Wide Web, pp. 661–670, 2010, https://doi.org/10.1145/1772690.1772758.

[4] M. Dudík, J. Langford, and L. Li, “Doubly robust policy evaluation and learning,” in Proceedings of the 28th International Conference on Machine Learning, pp. 1097–1104, 2011.

[5] T. Lattimore and C. Szepesvári, Bandit Algorithms. Cambridge University Press, 2020, https://doi.org/10.1017/9781108571401.

[6] D. Bouneffouf and I. Rish, “A survey on practical applications of multi-armed and contextual bandits,” arXiv preprint arXiv:1904.10040, 2019, https://doi.org/10.48550/arXiv.1904.10040.

[7] M. Gan and O.-C. Kwon, “A knowledge-enhanced contextual bandit approach for personalized recommendation in dynamic domains,” Knowledge-Based Systems, vol. 251, Art. no. 109158, 2022, https://doi.org/10.1016/j.knosys.2022.109158.

[8] A. Pilani, K. Mathur, H. Agrawald, D. Chandola, V. A. Tikkiwal, and A. Kumar, “Cont extual bandit approach-based recommendation system for personalized web-based services,” A pplied Artificial Intelligence, vol. 35, no. 7, pp. 489–504, 2021, https://doi.org/10.1080/08839514.2021.1883855.

[9] D. Glowacka, “Bandit algorithms in recommender systems,” in Proceedings of the 13th ACM Conference on Recommender Systems, pp. 574–575, 2019, https://doi.org/10.1145/3298689.3346956.
[10] E. Gangan, M. Kudus, and E. Ilyushin, “Survey of multiarmed bandit algorithms applied to recommendation systems,” International Journal of Open Information Technologies, vol. 9, no. 4, pp. 12–27, 2021.