Comprehensive Exploration and Implementation of Multi-Armed Bandit Algorithms Across Various Domains

Junyang Liu *

College of Engineering, China Agricultural University, Beijing, 100083, China

* Corresponding Author Email: 20203620@stu.hebmu.edu.cn

Abstract. This paper presents a comprehensive analysis of Multi-Armed Bandit (MAB) algorithms, elucidating their decision-making mechanisms in uncertain environments and their widespread applications in fields such as recommendation systems, advertisement delivery, network traffic control, and medical experiment design. Despite the notable successes of MAB algorithms, they encounter significant challenges in practical deployment, including the balance between exploration and exploitation, addressing unfairness, and managing large-scale implementations. An in-depth theoretical and practical examination of MAB algorithms is thus both theoretically and practically vital. This study conducts a meticulous literature review, theoretical analysis, and empirical research to offer insights into the current state and future prospects of MAB algorithms. It begins with a literature review, mapping out the research landscape and developmental trajectory of MAB algorithms. This is followed by a theoretical dissection, delving into the fundamental theories and diverse applications of MAB algorithms, with a focus on their utilization in recommendation systems, vehicle edge computing, and taxi route recommendation systems. Empirical research is then employed to validate proposed solutions and enhancement strategies. The study reinterprets the internal mechanics of MAB algorithms, affirming their effectiveness and superiority across various domains. The final section addresses the practical challenges and issues faced by MAB algorithms, such as fairness concerns and scalability. Corresponding solutions and improvement strategies are suggested, aiming to enhance the efficiency and applicability of MAB algorithms in real-world scenarios.

Keywords: Multi-Armed Bandit algorithms; Reinforcement learning; Application research; Fairness; Multi-objective optimization; Model variant.

1. Introduction

Multi-Armed Bandit algorithms represent a quintessential decision-making challenge, where an agent must choose among several options, or "arms", each with its own uncertain reward [1]. The core concept of MABs is rooted in the exploration-exploitation dilemma, seeking to maximize cumulative rewards by balancing the act of sampling different arms for information against exploiting the one expected to offer the highest return. The versatility and effectiveness of these algorithms have led to their widespread application in various fields.

In computer science, MAB algorithms are pivotal for dynamic and adaptive decision-making processes. For instance, in fuzz testing, they enhance the efficacy of test case selection, thereby uncovering software vulnerabilities more proficiently. Furthermore, multi-agent adaptations of the traditional MAB problem have been developed, facilitating collaborative learning among multiple agents with potentially divergent goals [2].

MAB algorithms have transcended the boundaries of computer science, marking their presence in other sectors. In clinical trials, they assist in refining treatment allocation strategies to optimize patient outcomes while mitigating risks. Additionally, in recommendation systems, MABs are instrumental in tailoring content by continuously adapting to user preferences [3]. Another area benefitting from these algorithms is adaptive routing, where they contribute to dynamic adjustments in routing strategies to enhance overall performance. The historical significance and extensive applicability of MAB algorithms highlight their crucial role in navigating complex decision-making scenarios across a range of domains. As these algorithms evolve within the active learning framework, their influence in devising solutions for intricate industry challenges is anticipated to grow [4].

The invention of the sophisticated Multi-Armed Bandit algorithm has opened doors to its application in a multitude of fields. Challenges such as maximizing user engagement in recommendation systems and optimizing multi-element processing in onboard edge computing networks are now being addressed. In taxi routing systems, MABs are being used to minimize costs and expedite customer service [5]. However, with increasing application complexity, optimizing the algorithm itself has become an imperative task [6]. Issues such as inequity in decision-making require strategies that not only maximize gains but also consider fairness. This necessitates the inclusion of fairness constraints in the model. Moreover, the challenge of managing large-scale data in scenarios where decision-makers face numerous options also demands adaptations to the MAB model, enabling it to tackle a broader spectrum of problems effectively.

2. Theoretical Foundations

2.1. Understanding Multi-Armed Bandit Algorithms

To understand Multi-Armed Bandit algorithms, it is important to first understand what reinforcement learning is. In traditional machine learning, the machine simply passively takes samples and does not adjust the training model based on the samples it has already collected. Reinforcement learning, on the other hand, is mainly applicable in an evolving environment, where it can actively choose its own actions. And according to the different feedback received by the action to adjust their next action. In order to achieve a long-time better feedback quality. It just so happens that Multi-Armed Bandit algorithms can be applied to reinforcement learning. Multi-Armed Bandit are essentially games of chance. Suppose that there are K slot machines, and each machine has a probability of winning Pi, then there is a probability of (1-Pi) not winning. How can you maximize your expected return when you only have T chances? In the algorithm, the Bayesian total probability formula is applied [7]. The posterior distribution of Pi is adjusted based on the feedback from each pull. Finally, select the slot machine with the posterior Pi distribution to continue pulling.

2.2. Operational Mechanics of Multi-Armed Bandit Algorithms

2.2.1. Explore-Then-commit Policy

The basic idea of the ETC Policy algorithm is a trade-off between Exploration and Exploitation. The steps are as follows. Initialize the counter for each arm and the cumulative return is 0. Select a random arm with probability ‘epsilon’; Select the arm with the highest current cumulative return with probability ‘1-epsilon’. Execute its selected arm and record its return [8]. Update the counter and cumulative return of the selected arm. Repeat steps 2-4 until time stops.

Although this algorithm is easy to understand and implement, it may not be balanced between Exploration and Exploitation, resulting in poor results.

2.2.2. Thompson Sampling Policy

The core idea of Thompson Sampling algorithm is to estimate the probability distribution of each arm by using Bayesian formula. Each time the arm is selected, it is sampled according to a posterior probability distribution. This makes the algorithm achieve a balance between Exploration and Exploitation, and performs well in practice [9]. The steps are as follows. Initialize a probability distribution Pi for each arm ‘I’. For each time step‘t’, a sample ‘ri’ is taken for each arm ‘I’ from its probability distribution ‘Pi’. Select the arm with the highest sampling value ‘ri’ as the current selection. Execute the selected arm and record its return. Update the probability distribution ‘Pi’ for each arm based on the results of the execution. Repeat Steps 2-5 until the time step is complete.

Thompson Sampling algorithm performs well in practice, especially in steady state environment. It can adapt to different probability distributions while avoiding the shortcomings of the algorithm in ETC Policy.

2.2.3. Upper Confidence Bound Policy

UCB algorithm is a greedy algorithm to solve the problem of Multi-Armed Bandit machines. The basic idea of the algorithm is to select the arm with the highest confidence by calculating the confidence interval of each arm [10]. The steps are as follows. Initialize the counter for each arm and the cumulative return is 0. Perform a select and perform operation once for each arm 'I' and record the return 'ri'. Update the counter and cumulative return of the selected arm 'I'. For each time step t, the confidence interval last 'UCB-i(t)' for each arm 'I' is calculated. Select the arm with the highest UCB value as the current selection. Repeat Steps 2-5 until the time step is complete.

UCB algorithm is a simple and effective algorithm that performs well in practice. It balances Exploration and Exploitation while being able to accommodate different probability distributions. However, it is sensitive to the selection of hyperparameters and needs to be carefully adjusted.

3. Application Research in Varied Domains

3.1. Multi-Armed Bandit Algorithms in Recommendation Systems

In a recommendation system, two competing goals need to be balanced: maximizing user satisfaction using spending history while gathering new information to improve the match between user preferences and items. The core idea of Multi-Armed Bandit machine algorithm in recommendation system is to use Bandit algorithm to balance exploration and utilization problems. Specifically, when the Multi-Armed Bandit machine algorithm is applied in the recommendation system, it will face two main problems: First, how to use the historical behavior data of the user to predict the interests and preferences of the user. The second is how to strike a balance between exploring new projects and using existing information to get the most out of them. Bandit algorithm can better balance the problem of "Exploration and Exploitation", without accumulating a lot of data in advance, it can better deal with the problem of cold start, avoid the Matthew effect caused by direct income, and avoid most long-tail and new resources without any opportunity to display.

Specific application steps are as follows:

Define the recommendation task: First of all, it is necessary to determine the objectives and constraints of the recommendation task, such as the type of recommended items, the time and place of recommendation.

Collect user data: Collect historical user behavior data, such as browsing history, purchase history, rating history, etc., to understand user interests and preferences.

Design recommendation strategies: Design appropriate recommendation strategies based on user data and recommendation tasks, such as content-based recommendation, social network-based recommendation, heat-based recommendation, etc.

Model training: The Multi-Armed Bandit machine algorithm is used to train the recommendation strategy to optimize the recommendation effect. Specifically, you need to define a reward function and an exploration strategy, and increase the value of the reward function by constantly adjusting the policy parameters.

Model application: The trained model is applied to the actual recommendation system to generate personalized recommendation results according to the user's behavior data and context information.

Feedback and adjustment: According to the feedback of users and the performance of the system, the recommendation strategy and model are constantly adjusted and optimized to improve the accuracy of recommendation and user satisfaction.

The application of Multi-Armed Bandit machine algorithm in the recommendation system can help the system better understand user needs and behaviors, improve the accuracy of recommendation and user satisfaction, and thus improve the overall performance of the system.

3.2. Implementation in Vehicle Edge Computing Systems

In recent years, the development of the Internet of Things and wireless technology has made continuous progress in the field of car networking, and cutting-edge applications such as autonomous driving and augmented reality are emerging. It is predicted that by 2025, the number of vehicles and on-board devices on the road will reach nearly 2 billion, with each vehicle generating up to 30 terabytes of data per day. Then how to deal with these high, heterogeneous data volumes becomes the next challenge.

In the present study, vehicle computing resources are considered for task offloading by diverting computing resources from the RSU equipped with MEC server to the on-board device layer. On the one hand, it eases the workload of the MEC server, on the other hand, the on-board equipment or passengers can take advantage of nearby vehicles with idle computing resources to perform task offloading. The traditional Multi-Armed Bandit machine framework is adopted in designing the task unloading algorithm. The traditional MAB problem is a trade-off between Exploration and Exploitation, where the player explores every option in the action set, which has a different distribution of rewards, and then uses the information learned to combine Exploration and Exploitation to choose the best action empirically.

Based on the MAB theory, a UCB based second-order exploratory reinforcement learning algorithm is proposed, which is executed by the user and provides the user with unloading decisions to maximize the average unloading return of the user. The algorithm can work independently, does not require additional signaling interaction with R, and can learn to assist in calculating the vehicle's service performance during task offloading. The computational complexity of this algorithm is low and it is unnecessary information interaction with other vehicles is easy to implement in the actual vehicle network.

3.3. Innovations in Taxi Route Recommendation Systems

In recent years, with the improvement of the transportation network and the popularity of online car booking, taxi travel has become the first choice of passengers. But the city's traffic is clearly tidal. During rush hours, the roads are congested and taxis can't pick up passengers quickly. In the normal peak period, the taxi is too far away from the passengers. Therefore, it is hoped that the path recommendation for the taxi can make the taxi have the maximum probability to pick up potential passengers in the next path.

The UCB algorithm in Multi-Armed Bandit plays a huge role in path recommendation. Together with MCTS, path recommendation results for road traffic have made a major breakthrough in time and space search speed and efficiency.

In the offline learning model, the historical passenger request located near the road is learned and the UCB value is obtained, which can be applied to the process of taxi dynamic path recommendation. In the online recommendation stage, the taxi, as a query node, recommends the maximum UCB value to the node and generates a new recommendation result for the next taxi cruise.

Step 1 set a real taxi as the query node.

Step 2 Obtain the UCB value near the node in the offline learning phase.

Step 3 Proceed to the road with the highest UCB value.

Step 4 if the taxi does not receive a passenger, perform Step 1 and perform negative feedback at the same time, using MCTS to update all UCB values of the taxi on the road to the offline learning system.

Step 5 if the taxi receives the passenger, complete the query.

The above steps are represented by the algorithm pseudo-code as follows:

Enter: Initial node R of path p.

Output: ρUCBp of the child imaxUCB of path p.

1. Set the UCB value ρUCBp = 0 for all roads in the network.
2. Set the initial value of round r to r = 0.
3. Loop: For each path p, set Si = 0.

4. Limit the total number of searches t in the set range [tmin, tmax] in.
5. Based on UCB algorithm and equation (3):
6. If ∀S'i ϵ RSi, i.e. the passenger request is found, then:
7. Update ρUCBp on all roads traversed.
8. Otherwise, update directly.

4. Challenges and Future Directions

4.1. Addressing Key Challenges

Artificial intelligence and Machine Learning algorithms are widely used to help people make decisions in their daily lives. However, with the popularity of algorithm application, problems such as big data killing and exposure bias have become increasingly prominent, and social individuals or groups have unfair phenomena in resource ownership, allocation and use, which will greatly affect user satisfaction and user trust in the algorithm, and may even lead to adverse social impacts. Algorithms are designed by humans, and algorithmic decisions may be inherently prone to unfairness, and biased data will also lead to unfairness of algorithms.

Most of the goals of machine learning algorithms are relatively simple, generally to minimize cumulative losses, which is equivalent to maximizing training accuracy and maximizing cumulative benefits. However, only considering to help decision makers maximize benefits usually leads to "fairness crisis". In 2013, for example, Latanya Sweeney, a professor of government and technology at Harvard University, published a paper showing the racial discrimination implied by Google's ad-serving algorithm. In 2019, the U.S. Department of Housing and Urban Development sued Facebook for pushing ads to specific users based on gender, race, religion and other attributes, and the algorithm learned from historical data that it can bring in more revenue by showing men ads for selling homes compared to women, which undoubtedly leads to gender discrimination.

In addition, in practical applications, the options faced by decision makers usually have a large number of scales, and multiple options need to be selected in a combined manner. Therefore, in some special cases, the model of the Multi-Armed Bandit needs to be deformed to cope with a variety of environments.

4.2. Future Research and Development Perspectives

Improvement of ε-Greedy algorithm: ε-Greedy algorithm is a commonly used algorithm for solving Multi-Armed Bandit problems, but its performance is not always optimal. It is an important direction to study how to improve the ε-greedy algorithm to improve its performance. For example, consider introducing more complex strategies, such as using randomization or increasing the frequency and amplitude of exploration, to better balance exploration and utilization.

Dynamic multi-arm slots: In real life, the rewards and probabilities of multi-arm slots may change over time. It is also an important direction to study how to design effective algorithms in this dynamic environment. For example, consider using adaptive algorithms to track changes in rewards and probabilities and adjust strategies accordingly.

Multi-agent Multi-Armed Bandit: When there are multiple Multi-Armed Bandits to choose from, how to design effective algorithms to choose the best strategy is also an important direction. We can consider using Multi-Agent learning methods, such as multi-agent Q-learning, to solve this problem.

Empirical research: Finally, the practical application and empirical research of multi-arm slot machine algorithm is also an important direction. The performance of an algorithm can be evaluated by testing its effectiveness in an experimental environment, or by deploying the algorithm in a real-world environment and observing its performance. In addition, the superiority of the proposed algorithm can be evaluated by comparing it with other algorithms or benchmarks.

5. Conclusion

This study conducts an in-depth examination of the Multi-Armed Bandit algorithm, both in theoretical and practical domains, affirming its efficacy and advantages across various fields. Concurrently, it scrutinizes the challenges and dilemmas encountered in the real-world application of the MAB algorithm. By identifying these issues, the paper proposes relevant solutions and enhancement strategies, offering insightful guidance for the ongoing research and practical deployment of the MAB algorithm. This comprehensive approach not only underscores the algorithm's versatility but also paves the way for its refined application and future advancements, thereby contributing significantly to the field.

References

[1] Mattos, D. I., Bosch, J., & Olsson, H. H. (2019). Multi-armed bandits in the wild: Pitfalls and strategies in online experiments. Information and Software Technology, 113, 68-81.

[2] Galichet, N. (2015). Contributions to multi-armed bandits: Risk-awareness and sub-sampling for linear contextual bandits (Doctoral dissertation, Université Paris Sud-Paris XI).

[3] Leqi, L., Zhou, G., Kilinc-Karzan, F., Lipton, Z., & Montgomery, A. (2023, April). A Field Test of Bandit Algorithms for Recommendations: Understanding the Validity of Assumptions on Human Preferences in Multi-armed Bandits. In Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems (pp. 1-16).

[4] Carrascosa, M., & Bellalta, B. (2020). Multi-armed bandits for decentralized AP selection in enterprise WLANs. Computer Communications, 159, 108-123.

[5] Wang, Q., & Grace, D. (2023). A Hybrid Proactive Caching System in Vehicular Networks Based on Contextual Multi-Armed Bandit Learning. IEEE Access, 11, 29074-29090.

[6] Zhu, X., Xu, H., Zhao, Z., & others. (2021). an Environmental Intrusion Detection Technology Based on WiFi. Wireless Personal Communications, 119(2), 1425-1436.

[7] Darak, S. J., & Hanawal, M. K. (2019). Multi-player multi-armed bandits for stable allocation in heterogeneous ad-hoc networks. IEEE Journal on Selected Areas in Communications, 37(10), 2350-2363.

[8] Liu, X., Zhu, T., Jiang, C., Ye, D., & Zhao, F. (2022). Prioritized experience replay based on multi-armed bandit. Expert Systems with Applications, 189, 116023.

[9] Carrascosa Zamacois, M., & Bellalta, B. (2020). Multi-armed bandits for decentralized AP selection in enterprise WLANs. Computer Communications. 2020 Jun 1; 159: 108-23.

[10] Schumann, C., Counts, S. N., Foster, J. S., & Dickerson, J. P. (2017). The diverse cohort selection problem: Multi-armed bandits with varied pulls. CoRR, abs/1709.03441.