# Bandit Algorithms for Advertising Optimization: A Comparative Study

*Ziyue* Tian*

School of Computer Science, Wuhan University, 430072, Wuhan, Hubei, China

**Abstract.** In recent years, the rapid development of digital advertising has challenged advertisers to make optimal choices among multiple options quickly. This is crucial for increasing user engagement and return on investment. However, traditional A/B testing often suffers from slow response times and difficulties in adapting to dynamic environments, leading to limited effectiveness. This article explores the application of multi-armed bandit algorithms in digital advertising, focusing on the performance of $\varepsilon$-greedy, Upper Confidence Bound (UCB), Linear Upper Confidence Bound (LinUCB), and Softmax algorithms. Experimental results show that incorporating user characteristics can significantly improve the accuracy and relevance of advertising recommendations. Among the algorithms tested, LinUCB, which utilizes contextual information, outperformed the other three non-contextual algorithms in terms of cumulative return and accuracy. It demonstrated significant advantages after full exploration. However, a limitation of this study is that the static experimental dataset cannot fully simulate the dynamic feedback of real-time advertising environments. This limitation affects the generalizability of the findings in highly changing contexts. Future research should focus on developing algorithms that can address feedback delays and are both adaptive and context-sensitive. This would enhance performance in complex advertising situations. Overall, this study deepens the understanding of multi-armed bandit algorithms in advertising and provides strategic guidance for user-targeted advertising.

## 1 Introduction

With the rapid development of the Internet, advertising placement has become increasingly complex in today's highly competitive advertising environment. Advertisers not only need to choose appropriate advertising content, but also determine the optimal audience group, and make quick decisions among numerous advertising options to obtain maximum user engagement and return on investment. However, advertising performance depends on multiple factors, such as ad content, display time, and target audience characteristics, which are often uncertain and dynamically changing. In this situation, optimization of advertising becomes particularly important. For traditional methods such as A/B testing, although some solutions are provided, they often require longer testing times and can result in significant costs [1].

* Corresponding author: 2023302111355@whu.edu.cn

To address these challenges, the Multi-Armed Bandit (MAB) algorithm has become an important tool in the field of advertising optimization in recent years. MAB algorithm can strike a balance between exploring new advertising options and utilizing existing data, helping advertisers continue to optimize advertising strategies in an uncertain environment [2]. Compared with traditional A/B testing, the MAB algorithm has higher real-time adaptability and resource utilization efficiency, so it has been widely used in the advertising field [3].

In the research field of combining multi-armed bandit algorithms with advertising placement, although many achievements have been made, there are still some unexplored research gaps and unsolved problems. First of all, most current multi-armed bandit algorithms are based on single-dimensional user behavior data (such as click-through rate, and browsing history), but advertising affects not only depend on these data, but also the user's emotions, location, social relationships, and time closely related to many factors [4]. How to integrate these multidimensional data into algorithms is still a problem that requires in-depth study [5]. Secondly, how to reduce estimation bias in practical applications is also a key issue. By identifying more effective algorithm strategies to accurately estimate the potential benefits of options, the selection of suboptimal options can be minimized, thereby optimizing decision-making outcomes [6]. In addition, the combination of reinforcement learning and deep reinforcement learning, especially its application in multi-armed bandit algorithms, is still an area that requires in-depth research. Existing research mainly focuses on how to improve the learning efficiency and adaptability of the model, but how to effectively shorten the training cycle of the model is also a topic worthy of in-depth study [7].

## 2 Related work

The MAB problem originated from the fields of probability theory and decision theory. Its core is how to make optimal decisions among multiple choices and strike a balance between exploration and exploitation to obtain maximum benefits. In 1985, Lai and Robbins first raised this issue. It also proposed a progressively optimal adaptive allocation rule and constructed an early framework of the MAB algorithm, thus solving the optimal strategy problem of resource allocation under uncertainty and laying the foundation for the subsequent development of the MAB algorithm [8].In 1989, Gittins further expanded this theory and proposed the Gittins exponential method, which provided an effective solution for dynamic decision-making in uncertain environments to solve complex multi-armed bandit problems [9].In 2002, Auer et al. proposed the UCB algorithm to strike a balance between exploration and exploitation by using statistical confidence intervals to estimate the potential benefits of each option [10]. This landmark contribution promoted MAB's further development. With the rise of the Internet and digital advertising, MAB algorithms are gradually introduced into the advertising field to optimize advertising delivery. In this context, Schwartz et al. proposed a framework combining deep Q learning with the MAB algorithm in 2017 to achieve more efficient resource allocation in complex advertising environments [11]. Because this method combines the powerful functions of deep learning, the MAB algorithm can adapt in real-time to dynamically changing user environments, thus greatly improving the ability to optimize advertising effects [12]. At this point, the MAB algorithm has developed to a new level. In general, the MAB algorithm has experienced rapid development from theory to practical application and has become an indispensable tool in advertising and other decision-making fields. Future research will continue to explore how to improve these algorithms to cope with increasingly complex market environments and changes in user needs [13].

# 3 Methodology

The method in this study mainly consists of three steps, as shown in Figure 1, that are closely related. In the initial stage, the data is cleaned and processed to lay the foundation for subsequent stages of research. Then, based on the theoretical framework of the algorithm, models are constructed for the four main algorithms in MAB: ε-greedy algorithm, UCB algorithm, Thompson Sampling, and Softmax strategy. Finally, a multi-dimensional evaluation and horizontal comparison of these four MAB algorithms was conducted to evaluate their practical application and performance indicators in advertising.

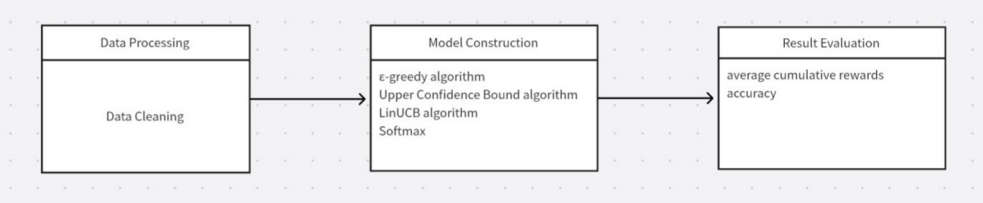


**Fig. 1.** The flow chart of this study

## 3.1 Data processing

The experimental data set of this study contains 10,000 rows, each row records the user's behavior in accessing the advertisement, and each row contains 102 columns of data. The first column represents different forms or types of advertisements (i.e., randomly selected arms), and the second column represents the user's click status, where 1 means click and 0 means no click. Columns 3 to 102 contain 100-dimensional contextual information related to each advertisement, and the characteristics of each arm are divided into 10 columns, reflecting the degree of matching between the content or form of the advertisement and the user. This contextual feature will be used in the algorithm to optimize arm selection.

## 3.2 Algorithm principles and optimization

### *3.2.1 ε-greedy algorithm*

The core idea of the ε-greedy algorithm is to utilize the current best options while still retaining a certain proportion of time for exploration to prevent premature falling into the local optimal solution. Because even if the current optimal solution seems to have the highest benefit, there may be other better actions that have not been fully explored. At each step of the algorithm, an action is randomly selected with probability ε (exploring to try to find an action with a higher reward), the action with the highest current estimated return is selected with probability 1-ε (thus maximizing short-term returns), and the estimate for each action is updated through gradually accumulated feedback data. It can be seen that the ε-greedy algorithm does not require complex mathematical calculations or model assumptions, so it is suitable for many practical scenarios that require simple decisions. However, since the exploration of the ε-greedy algorithm is completely random, this means that even if the return of a certain arm is low, the algorithm may still choose it, thus causing waste.

### *3.2.2 UCB algorithm*

The UCB algorithm introduces a new concept called confidence interval. The confidence

interval of each action consists of two parts: one is the historical average return value of the action, and the other is an exploration term that is inversely proportional to the number of times the action is selected. The UCB algorithm determines whether to select an action by calculating the upper confidence bound of each action. This allows UCB to not only take into account the average return of each action but also introduces uncertainty factors, making actions with larger confidence intervals (i.e., uncertain Actions that are more sexual) more likely to be selected for exploration. It can be seen that the UCB algorithm can automatically balance the relationship between exploration and utilization without manually setting the ratio. In addition, compared to ε-greedy's random exploration, UCB can more effectively select options that are not fully explored but have potentially high returns. However, since UCB needs to calculate the upper bound of the confidence interval of all actions at each time step, the calculation amount is large when the number of actions is large.

### *3.2.3 LinUCB algorithm*

LinUCB is an algorithm that extends UCB (Upper Confidence Bound) and is suitable for solving multi-armed bandit problems with contextual information or when dealing with linear reward models. Unlike classic UCB, LinUCB better estimates the potential reward of each action by leveraging contextual information (related to the environment, user characteristics, or the action itself). This contextual information helps the algorithm consider more comprehensive factors when making decisions, so that the algorithm does not just rely on past selection results, but makes more appropriate decisions based on the current context. It assumes that the reward of each action can be expressed as the inner product of contextual features and a set of unknown linear parameters plus some noise. Therefore, the algorithm needs to continuously update the parameters of the linear model through the contextual features and historical returns of each action to make more accurate decisions.

### *3.2.4 Softmax*

The core idea of the Softmax algorithm is to assign selection probabilities according to the expected return of each action, thereby making random selections based on these probabilities, rather than always selecting the action with the highest current return. Unlike ε-greedy, the Softmax strategy does not just binary select the best or random option but instead assigns the probability of selection based on the relative payoff of each action and utilizes the Softmax function. It can be seen from this that the Softmax strategy can allocate selection probabilities more naturally, and by adjusting the temperature parameter τ, the exploration intensity can be flexibly controlled.

## 4 Experiment and results

The paper explored the performance of four major multi-armed bandit algorithms, namely ε-greedy, UCB, LinUCB, and Softmax, in advertising, and analyzed the performance differences of each algorithm when having user context information. The dataset used in the experiment contains contextual features and reward data of users interacting with different advertising arms, where each advertising arm has unique contextual features. Observe how well each algorithm performs in optimizing ad selection by tracking its cumulative reward and accuracy over multiple rounds.

### 4.1 Parameter Settings

To test the adaptability of these algorithms in a dynamic advertising environment, all algorithms were run for 1000 rounds under different parameter settings to ensure that the differences in rewards brought by the balance between different exploration and exploitation can be fully demonstrated. Each ad arm has a different reward distribution, simulating the situation of real ad click-through rates. In each round, each algorithm chooses to explore or utilize different advertising arms based on the currently observed reward information and updates the arm information. Among them, LinUCB will select the advertising arm based on contextual information, and the α value is set to 1 to ensure that the algorithm conducts sufficient exploration and gradually converges to a better utilization strategy. In the UCB algorithm, ρ is set to 1, contextual information is not used, but the advertising arm is selected based on the upper confidence limit of the reward. The exploration rate ε of the ε-greedy algorithm is set to 0.05, which means that in each round of selection, there is a 5% probability of randomly exploring the advertising arm and the remaining 95% probability of selecting the advertising arm with the highest current reward. The temperature parameter τ of the Softmax algorithm is set to 0.1, so that at lower temperature values, the algorithm is more inclined to use existing reward information, but at the same time maintains a certain degree of exploitability.

### 4.2 Evaluation indicators

This experiment mainly evaluates the performance of the algorithm through cumulative rewards and accuracy. The cumulative reward represents the total reward obtained in each round of experiments. The higher the cumulative reward, the algorithm can better balance exploration and utilization, and identify and select the advertising arm with the highest return in a shorter time. The accuracy rate indicates how well the selected ad arm matches the actual highest reward ad arm. Reflects the algorithm's ability to identify ads relevant to user interests.

Figure 2 shows the average cumulative reward curves of LinUCB, UCB, ε-greedy, and Softmax algorithms during 10,000 rounds of experiments. It is not difficult to see from the figure that the LinUCB algorithm performed well during the experiment, and its cumulative reward is much higher than other algorithms.

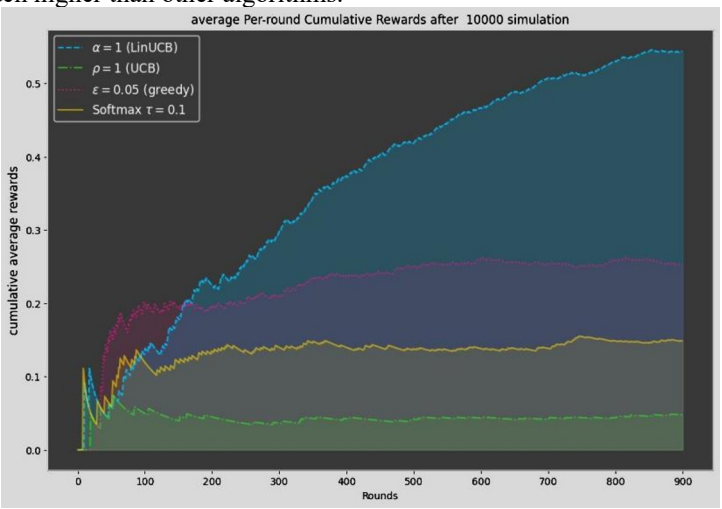


**Fig. 2.** Comparison of Cumulative Rewards by Exploration Strategy

This shows that LinUCB can combine contextual data well and achieve a good balance

between exploration and exploitation, thereby efficiently and continuously select the advertising arm with the highest potential revenue. Further observation found that although the UCB algorithm can quickly accumulate rewards in the initial stage, as the experiment progresses, its average reward does not rise but falls, and its performance is inferior to the other three algorithms. The reason may be that it relies too much on historical data and ignores new information. For the ε-greedy algorithm, higher rewards were obtained by extensively trying various advertisements in the early and middle stages of the experiment. However, this exploration method reduces the learning efficiency in the later period and wastes choices on advertising arms with low returns. Therefore, as the experiment progresses, the average reward of the algorithm in the later period is not as good as the LinUCB algorithm. In contrast, the Softmax algorithm provides a smoother learning process. Although it is not as efficient as LinUCB overall, it can maintain a relatively stable return.

To further analyze the performance of different strategies over time, Figure 3 shows the comparison of the accuracy of LinUCB and UCB at different times.

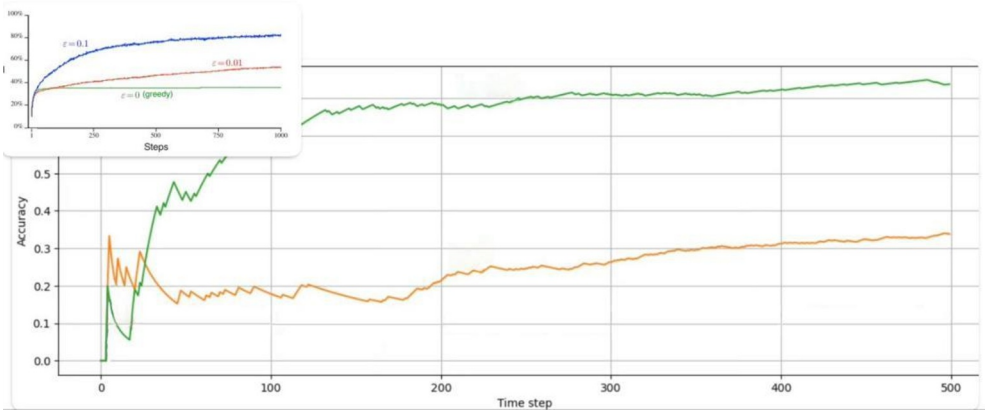


**Fig. 3.** Comparison of Accuracy between LinUCB and UCB over Time Steps

In the first 100-time steps, the two algorithms fluctuated but generally showed an upward trend. After that, the accuracy of LinUCB began to significantly surpass UCB, and as the experiment progressed, the advantage continued to expand. This shows that LinUCB can better adapt to changes in the environment and more accurately select ads with the highest potential returns. In addition, to compare the impact of different ε values on the performance of the ε-greedy algorithm, the inserted small figure shows the accuracy performance under ε=0.1, ε=0.01 and the completely greedy strategy (ε=0). When ε=0, a completely greedy strategy is adopted. At this time, the strategy lacks necessary exploration and it is difficult to discover new options that may bring higher returns, so the algorithm has the lowest accuracy. As the ε value increases, the accuracy of the algorithm also increases. This phenomenon shows that balancing the ratio of exploration and utilization can effectively improve the accuracy of the algorithm.

## 5 Limitations and future outlooks

Although this article provides insights into the multi-armed bandit algorithm in advertising, it has some limitations. First, the dataset is too narrow. While it includes contextual features and user click data, it does not fully capture the complexity of advertising in a rapidly changing environment. In practice, advertisers face more diverse users, complex markets, and larger volumes of data, which limit the applicability of the findings.

Second, although the four classic multi-armed bandit algorithms discussed here have

shown results in advertising, they struggle in more complex situations. Traditional algorithms lack the flexibility and learning capabilities of deep learning, making them less effective in processing multi-dimensional feedback data. Additionally, this study did not fully consider potential fluctuations in user behavior, market competition, and national policies. It assumes a static environment, which simplifies the problem but does not reflect the need for algorithms to adapt to dynamic conditions.

Finally, the delay in advertising feedback is a limitation. There is often a time lag between a user's click and their purchase, which can interfere with the algorithm's decision-making. Future research should focus on optimizing algorithms to address the challenges posed by feedback delays. Looking ahead, improvements in algorithm performance should come from two key areas: enhancing the ability to process complex information and adapting to real-time changes. Research on advertising delivery algorithms should involve larger, more complex datasets and deep learning methods to improve the handling of multi-dimensional information. Additionally, developing algorithms with strong adaptive capabilities to manage changes in the delivery environment will be a crucial future research topic.

## 6 Conclusion

This study compares the performance of four algorithms—ε-greedy, UCB, LinUCB, and Softmax—in advertising scenarios. The results demonstrate LinUCB's superiority in handling complex decision-making situations. After sufficient rounds of exploration, LinUCB's ability to exploit contextual features leads to significantly higher returns compared to the other algorithms. In contrast, the performance of the ε-greedy algorithm declines as exploration frequency increases, revealing its instability. The UCB algorithm's failure to utilize contextual features limits its effectiveness in personalized advertising. While the Softmax strategy avoids the randomness of the ε-greedy algorithm, it converges slowly. LinUCB's capacity to understand and act on user characteristics makes it the best choice for personalized recommendations or targeted advertising. This study explores the application of four classic multi-armed bandit algorithms in advertising and experimentally verifies their performance in these scenarios, highlighting LinUCB's advantages in integrating contextual information. It underscores the importance of aligning advertising content with users' personalized needs in optimization algorithms.

## References

1. R. Kohavi, D. Tang, M. Ye, Controlled experiments on the web: survey and practical guide. Data Min. Knowl. Discov. 18, 3-18 (2009).
2. S.L. Scott, A primer on the use of multi-armed bandits in marketing. Mark. Res. 22, 12-19 (2010).
3. L. Li, S. Wang, H. Zheng, Contextual bandits with continuous actions. Proc. 27th Int. Conf. Mach. Learn. (ICML) (2010).
4. S.J. Pan, Q. Yang, A survey on transfer learning. IEEE Trans. Knowl. Data Eng. (2010).
5. Y. Wang, C. Liu, L. Zhang, Integrating multi-dimensional data in bandit algorithms. J. Mach. Learn. Res. 22, 1-30 (2021).
6. M. Kasy, S. Athey, Regression and inference in econometrics. Rev. Econ. Stud. 86, 1-31 (2019).
7. X. Chen, J. Huang, Y. Li, Accelerating training for multi-armed bandit algorithms. Artif. Intell. Rev. 54, 1-20 (2021).

8. T.L. Lai, H. Robbins, Asymptotically efficient adaptive allocation rules. Adv. Appl. Math. 6, 4-22 (1985).
9. J.C. Gittins, Bandit processes and dynamic allocation indices. J. R. Stat. Soc.: Ser. B (Methodol.) 41, 148-177 (1989).
10. P. Auer, N. Cesa-Bianchi, P. Fischer, Finite-time analysis of the multiarmed bandit problem. Mach. Learn. 47, 235-256 (2002).
11. G. Schwartz, L. Ward, N. Kallus, Deep Q-learning for multi-armed bandit problems. Proc. 34th Int. Conf. Mach. Learn. (ICML) (2017).
12. M. Kocak, et al., Real-time learning in digital advertising with MAB. Mark. Sci. 38, 304-319 (2019).
13. S. Agrawal, N. Goyal, Further results on bandit problems. Mach. Learn. 47, 235-256 (2013).