# Web Ads Optimization using Reinforcement Learning Techniques

**Vivek Singh[1], Charanjeet Singh[2]**
Student[1] Dept. of Electronics Engineering, DCRUST ,Murthal, Sonipat, India
Assistant Professor[2] Dept. of Electronics Engineering, DCRUST ,Murthal, Sonipat, India
xsvk01@gmail.com[1], dcrustcharanjeet@gmail.com[2]

**Abstract**: One of the crucial jobs in online business applications that involves internet advertising is click-through rate (CTR) prediction and recommendation system. These are the crucial components of online marketing platforms. Algorithms for machine learning are frequently used to overcome interaction issues. These challenges are particularly well suited for Reinforcement Learning methods. For better click-through rate prediction and ads recommendations, we have suggested a model in this paper based on the most suited Reinforcement learning techniques named Upper Confidence Bound and Thompson sampling. The dataset, which includes data on 10 advertisements, was created using Numpy and pandas python libraries that replicates the ads interaction on the online platform. On this dataset, we applied Upper confidence Bound and Thompson Sampling and an analysis of the results is also done which states which of the two algorithms is more suited according to the use cases depending on the interaction with the online web advertisements.

Keywords: Upper Confidence Bound (UCB), Thompson sampling, Reinforcement Learning (RL), Click-Through Rate (CTR), Machine Learning, Recommendation System (RS), Multi-Armed Bandits (MAB).

## I. INTRODUCTION

Search term advertising has risen to be a crucial component of online surfing. These adverts frequently pay for phrases through an allocation scheme. Both costs for hit and cost per click invoicing is used.

Reinforcement learning is the process of discovering how to connect events with behaviors observable [1]. Knowing the linkage and creating the cases are indeed the two fundamental components of RL (with the help of math models) [2], [3]. Multi-armed bandits and Markov decision processes are the two main environments for dilemma creation. A simple model for the exploration/exploitation trade-off has been related to the MAB [4].

The Partially Visible Markov decision process (PVMDP) extends the MDP to the situation in which the system state is not always observable [1] [4].

Such commercials, in the initial place, offer consumers advantages like product details, a substantial reduction, etc. Furthermore, these adverts undoubtedly add benefits to customers provided that now the intended population is targeted by the marketer' sales promotion [6]. Third, the search engine benefits from these advertisements as well since it will make money if consumers click on the adverts [6]. Thirdly, the SEO also gains from all these ads given that it will earn a living if users click on them.

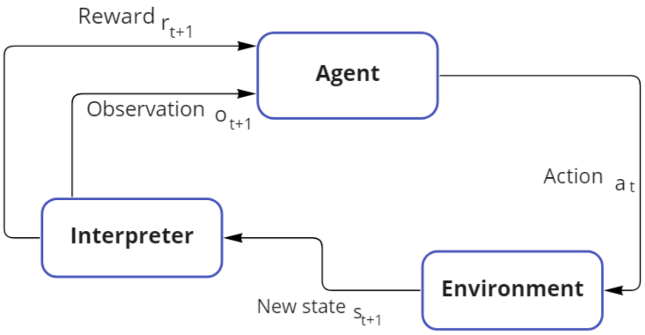


Fig 1. RL cycle. After assessing the environmental conditions, the agent attempts to react to a situation as effectively as possible. The probable payoff is converted together into state change brought about by the action and communicated to the unit together with a new assessment of the situation [5].

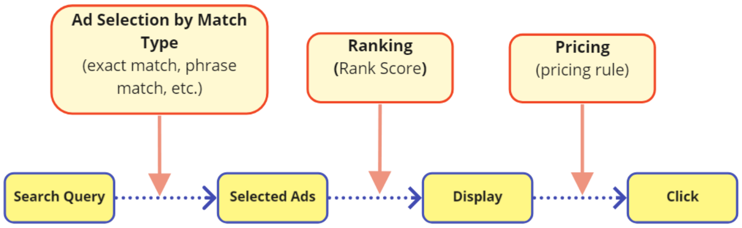


Fig 2. It is better to pick advertisements after closely examining every of these possible consequences, such as if the business selection approach may improve user interactions, advertiser support, and online search income [7].

Many marketers that are prepared to market online businesses have become interested in online marketing campaigns [8], [9]. Another of the major issues that advertisers, particularly individuals who do not have much experience in internet advertising, encounter is how to better structure their operations. Selecting the appropriate target is essential for creating the greatest game, which will ensure that people will accept your adverts on a good degree. Additionally, the quantity of visits needed to set up must be sufficient to support advertising efforts [10], [11].

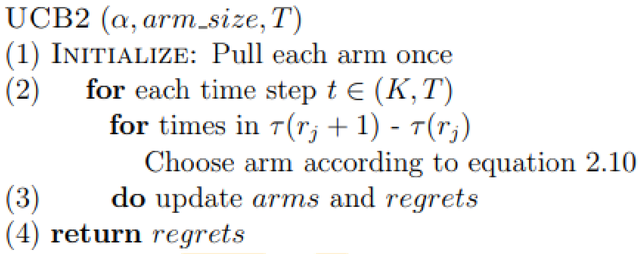
The confidence interval seen between perceived reward and the true reward is used by the Upper Confidence Bound approach to determine which of the K arms seems to be the best [12]. The confidence level depicts unpredictability; the wider the interval, the less guaranteed the reward for the arm is. Additionally, as the sequence goes longer, the agent is able to see more, and indeed the interval gets shorter. Ideally, the gap will finally take on a given value with an unlimited quantity of samples [12].

$$Q(a) \leq Q_t(a) + U_t(a)$$

Another method is to reduce the variable that governs the investigation as time grows, such as e - greedy, and prevent wasteful exploring where the agency could investigate activity that is already recognized to also be poor. A different strategy is used by UCB [13], which is poised to be upbeat and favors the choices with high levels of uncertainty. The agent will tend to favor acts that stand a good chance of producing the best results. An upper confidence bound also describes this [14]. At each time step, UCB will choose the arm with the maximum upper confidence bound Ut(a) [12].

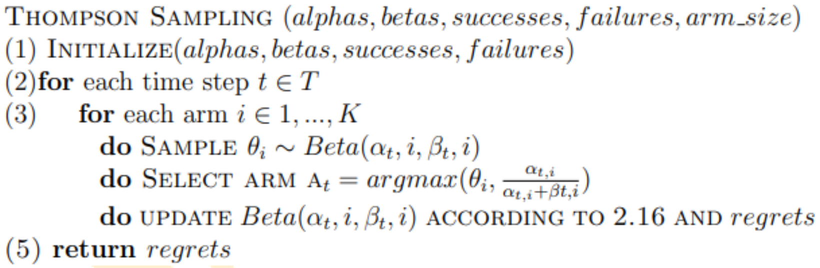
$$a_t = argmax_{a \in \mathcal{A}}(Q_t(a) + U_t(a))$$

Upper confidence Bound Pseudocode



```
UCB2 $(\alpha, arm\_size, T)$
(1) INITIALIZE: Pull each arm once
(2)     for each time step $t \in (K, T)$
          for times in $\tau(r_j + 1)$ - $\tau(r_j)$
             Choose arm according to equation 2.10
(3)       do update $arms$ and $regrets$
(4) return $regrets$
```

Thompson sampling Pseudocode



```
THOMPSON SAMPLING $(alphas, betas, successes, failures, arm\_size)$
(1) INITIALIZE$(alphas, betas, successes, failures)$
(2)for each time step $t \in T$
(3)     for each arm $i \in 1, ..., K$
          do SAMPLE $\theta_i \sim Beta(\alpha_t, i, \beta_t, i)$
          do SELECT ARM $A_t = argmax(\theta_i, \frac{\alpha_{t,i}}{\alpha_{t,i}+\beta t,i})$
          do UPDATE $Beta(\alpha_t, i, \beta_t, i)$ ACCORDING TO 2.16 AND $regrets$
(5) return $regrets$
```

## II. PROBLEM STATEMENT

The goal is to choose the most rewarding advertisement which has the highest click rate, to be shown more frequently which ultimately results in the increase in revenue.

The exploitation-exploration conundrum is a problem with traditional recommender systems. Exploration involves recommending items at random in order to gather more user feedback, whereas exploitation involves recommending items that are expected to best fit users' interests. The spatial bandit models an agent that attempts to strike a balance between the competing tasks of exploitation and research in maximizing the long term reward over a set period.

In a bandit situation, the conventional tactics to strike a balance between exploitation and exploration are "ε-greedy"[15], EXP3 [16], and UCB1 [16]. In order to boost the total amount of customer hits, a retraining method based LinUCB is presented to pick content systematically for individual users depending upon that background knowledge of the individuals and pieces. There in current feeds situation, the discovery challenge of tailored recommender system is described as a contextually bandit challenge [15].

Ten advertisements' worth of data are present in the dataset as prizes. These ten are all variations of the same advertisement [12], [17]. The finest advertisement to post on the social network must be identified in this case. The advertisement with the best click-through rate and most clicks will be placed. The challenge at hand is to determine which version of this advertisement is better for the user [7]. Only the simulation is included in this dataset. In the real world, we will begin testing by posting several versions of advertisements on social networks. Based on the outcomes, we will adjust our approach to posting these advertisements on social networks. We will display one of these 10 adverts each time a person logs into their social network account, and we'll monitor their behavior to see how they react. The reward is added if the user clicks the advertisement; else, the reward is 0. For a total of 10,000 users, this task is repeated.

## III. PROPOSED MODEL

The system will make use of an example dataset made with NumPy and Pandas. In addition, to sanitize the data, the exported dataset must go through a data reprocessing step. However, because the data was explicitly prepared using the Python module, cleaning is not necessary. Data pre-processing is a crucial procedure that ensures that the data is free of any errors or null values that just might significantly skew the findings. If there are any gaps in the data, then the mean or median can be employed to fill them. In addition, dependent on the dataset, feature set and label embedding are options that can be used. Then, we used the Thompson Sampling reinforcement method to determine whether it had any advantages over random selection. Then, using the dataset, we deployed Upper Confidence Bound technique and compared the outcomes. Python was chosen as the programming language for implementation.

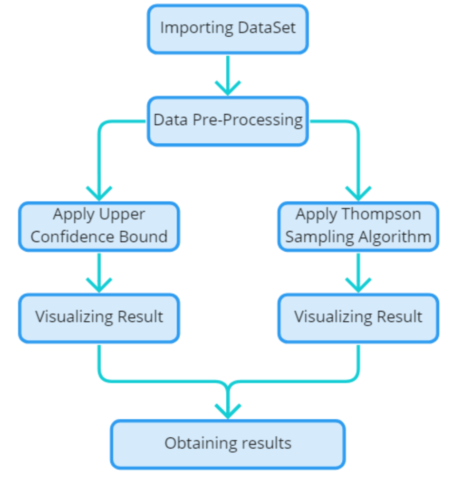
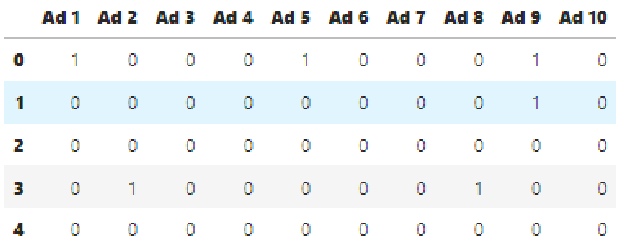


Fig 3. Flow chart for the implementation of Web Ad optimization techniques

The flow chart explains the use of the dataset after its detailed analysis and cleaning because having some invalid data in the dataset can cause either incorrect results or in the failure of the code execution. The algorithms used in this experiment are Upper confidence bound and Thompson's Sampling model. Then, in order to visualize the result, matplotlib library module is used in python and for visualizing the dataset, the JetBrains Datalore platform is used.

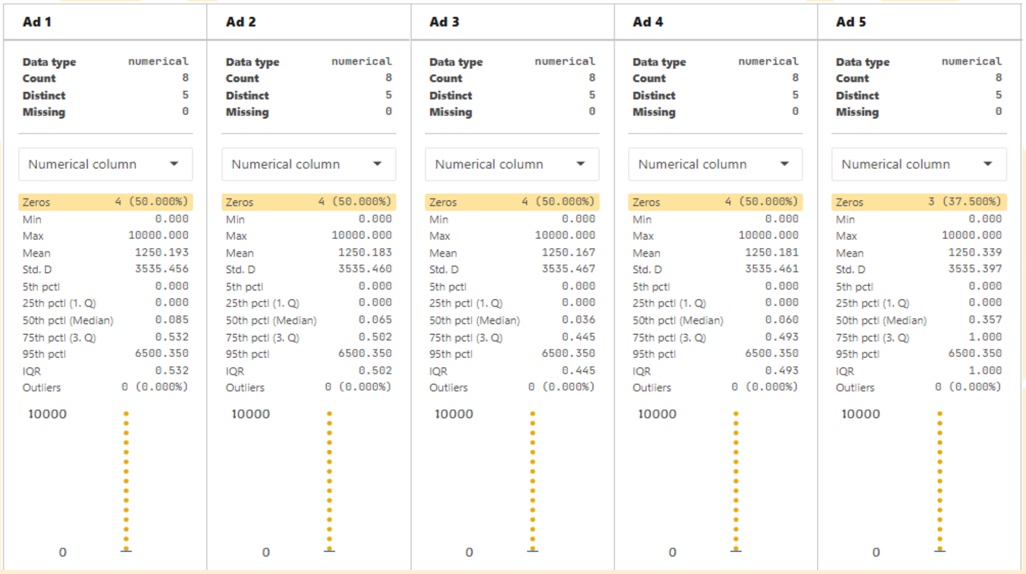
This dataset consists of a table with their column names as the specific Advertisement. The dataset is in csv format. Each row item represents the status of whether the ad was clicked or not, with '0' being 'not clicked' and '1' being 'clicked'. In order to access the contents of that data in Python, we can use panda's library and convert the .csv file into a data frame.



| | Ad 1 | Ad 2 | Ad 3 | Ad 4 | Ad 5 | Ad 6 | Ad 7 | Ad 8 | Ad 9 | Ad 10 |
|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| **1** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| **2** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **3** | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| **4** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

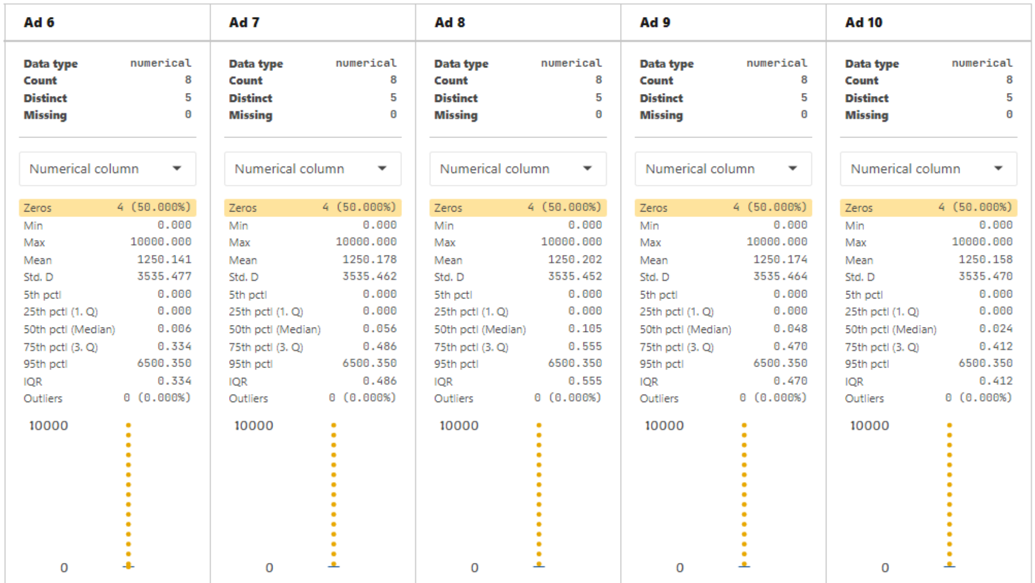
Fig 4. First 4 rows out of 10000 rows of the dataset.

The detailed description as showing in the Fig 5 and Fig 6 is shown using the Jet Brains datalore online platform, which analyze all the major aspects of the dataset table and shows the detailed description of each column.



| | Ad 1 | Ad 2 | Ad 3 | Ad 4 | Ad 5 |
|---|---|---|---|---|---|
| **Data type** | numerical | numerical | numerical | numerical | numerical |
| **Count** | 8 | 8 | 8 | 8 | 8 |
| **Distinct** | 5 | 5 | 5 | 5 | 5 |
| **Missing** | 0 | 0 | 0 | 0 | 0 |
| Zeros | 4 (50.000%) | 4 (50.000%) | 4 (50.000%) | 4 (50.000%) | 3 (37.500%) |
| Min | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 |
| Max | 10000.000 | 10000.000 | 10000.000 | 10000.000 | 10000.000 |
| Mean | 1250.193 | 1250.183 | 1250.167 | 1250.181 | 1250.339 |
| Std. D | 3535.456 | 3535.460 | 3535.467 | 3535.461 | 3535.397 |
| 5th pctl | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 |
| 25th pctl (1. Q) | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 |
| 50th pctl (Median) | 0.085 | 0.065 | 0.036 | 0.060 | 0.357 |
| 75th pctl (3. Q) | 0.532 | 0.502 | 0.445 | 0.493 | 1.000 |
| 95th pctl | 6500.350 | 6500.350 | 6500.350 | 6500.350 | 6500.350 |
| IQR | 0.532 | 0.502 | 0.445 | 0.493 | 1.000 |
| Outliers | 0 (0.000%) | 0 (0.000%) | 0 (0.000%) | 0 (0.000%) | 0 (0.000%) |

Fig 5. Detailed description of first 5 columns of the dataset



| | Ad 6 | Ad 7 | Ad 8 | Ad 9 | Ad 10 |
|---|---|---|---|---|---|
| **Data type** | numerical | numerical | numerical | numerical | numerical |
| **Count** | 8 | 8 | 8 | 8 | 8 |
| **Distinct** | 5 | 5 | 5 | 5 | 5 |
| **Missing** | 0 | 0 | 0 | 0 | 0 |
| | Numerical column | Numerical column | Numerical column | Numerical column | Numerical column |
| Zeros | 4 (50.000%) | 4 (50.000%) | 4 (50.000%) | 4 (50.000%) | 4 (50.000%) |
| Min | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 |
| Max | 10000.000 | 10000.000 | 10000.000 | 10000.000 | 10000.000 |
| Mean | 1250.141 | 1250.178 | 1250.202 | 1250.174 | 1250.158 |
| Std. D | 3535.477 | 3535.462 | 3535.452 | 3535.464 | 3535.470 |
| 5th pctl | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 |
| 25th pctl (1. Q) | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 |
| 50th pctl (Median) | 0.006 | 0.056 | 0.105 | 0.048 | 0.024 |
| 75th pctl (3. Q) | 0.334 | 0.486 | 0.555 | 0.470 | 0.412 |
| 95th pctl | 6500.350 | 6500.350 | 6500.350 | 6500.350 | 6500.350 |
| IQR | 0.334 | 0.486 | 0.555 | 0.470 | 0.412 |
| Outliers | 0 (0.000%) | 0 (0.000%) | 0 (0.000%) | 0 (0.000%) | 0 (0.000%) |

Fig 6. Detailed description of remaining columns of the dataset

## IV. RESULTS

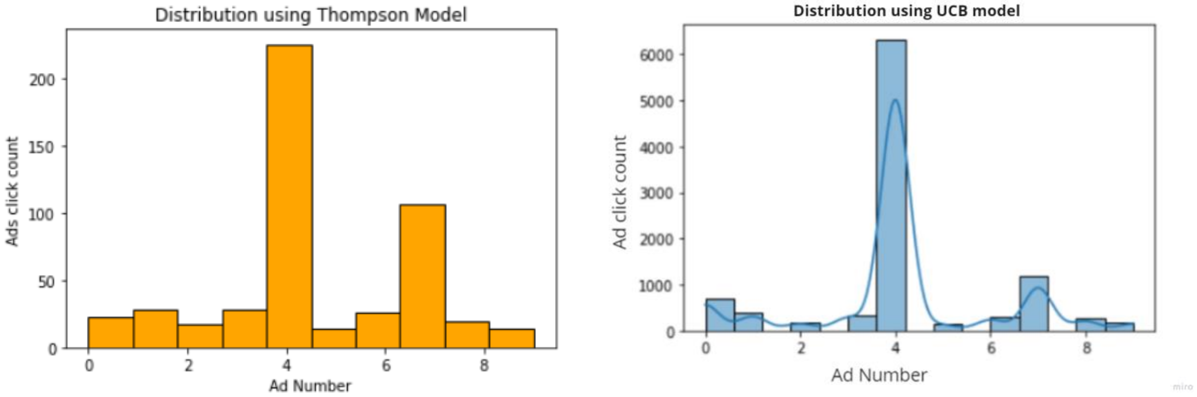


Fig 7. The comparison of the distributions of number of clicks on each ads recommended according to the algorithms used, i.e., “Thompson sampling” and “UCB” algorithm. The results are visualized with the help of the bar plots using the “matplotlib” module in Python.

From the above visualizations, we can see that ad number 4 got the highest click. Even though both of the algorithms tells to the user place the 4th ad more prominently for getting the highest number of clicks. When compared to randomized selection, and both UCB & Thompson Sampling techniques function. Several of the mathematical assurances of upper-confidence bound algorithms like Thompson sampling are based on a basic characteristic.

UCB tends to be more suitable for the case where the data and actions are less reliable as it calculates the confidence interval and uses the principle of optimism in the face of uncertainty. Also, as both algorithms are "optimistic" in the sense that they devote exploration effort to potential optimum behaviors, one may also integrate regret assessment over both of these techniques and several problem categories, as well as transfer regret bounds determined for UCB methods onto Bayesian regret bounds with Thompson sampling.

## V. CONCLUSION

Across every situation, Thompson Sampling and UCB proved to be able to achieve maximum overall reward despite keeping careful analysis of many variations and the capacity to spot variance among them. Thompson Sampling might be a better choice in circumstances where the method is more reliable, like those that have a higher lower bound conversion efficiency or estimated affect size. However, because of its reliability and high tolerance for data distortion, UCB is the best Multi Armed Bandit method in situations with relatively low conversion rates and minor effect sizes.

## VI. REFERENCES

[1] R. D. Smallwood and E. J. Sondik, "OPTIMAL CONTROL OF PARTIALLY OBSERVABLE MARKOV PROCESSES OVER A FINITE HORIZON.," *Oper Res*, vol. 21, no. 5, pp. 1071–1088, 1973, doi: 10.1287/opre.21.5.1071.

[2] G. Zheng *et al.*, "DRN: A deep reinforcement learning framework for news recommendation," in *The Web Conference 2018 - Proceedings of the World Wide Web Conference, WWW 2018*, Apr. 2018, pp. 167–176. doi: 10.1145/3178876.3185994.

[3] A. Mosavi *et al.*, "Comprehensive review of deep reinforcement learning methods and applications in economics," *Mathematics*, vol. 8, no. 10, Oct. 2020, doi: 10.3390/MATH8101640.

[4] E. J. Sondik, "OPTIMAL CONTROL OF PARTIALLY OBSERVABLE MARKOV PROCESSES OVER THE INFINITE HORIZON: DISCOUNTED COSTS.," *Oper Res*, vol. 26, no. 2, pp. 282–304, 1978, doi: 10.1287/opre.26.2.282.

[5] A. Mäkelä, "Deep reinforcement learning as a tool for search engine campaign budget optimization A dive into deep reinforcement learning and its application to optimizing budget allocation between search engine advertising campaigns Atte Mäkelä."

[6] Q. Cui, F. S. Bai, B. Gao, and T. Y. Liu, "Global Optimization for Advertisement Selection in Sponsored Search," *J Comput Sci Technol*, vol. 30, no. 2, pp. 295–310, Mar. 2015, doi: 10.1007/s11390-015-1523-4.

[7] L. Li, W. Chu, J. Langford, and R. E. Schapire, "A Contextual-Bandit Approach to Personalized News Article Recommendation," Feb. 2010, doi: 10.1145/1772690.1772758.

[8] R. R. Afshar, Y. Zhang, M. Firat, and U. Kaymak, "A reinforcement learning method to select ad networks in waterfall strategy," in *ICAART 2019 - Proceedings of the 11th International Conference on Agents and Artificial Intelligence*, 2019, vol. 2, pp. 256–265. doi: 10.5220/0007395502560265.

[9] D. Rohde, S. Bonner, T. Dunlop, F. Vasile, and A. Karatzoglou, "RecoGym: A Reinforcement Learning Environment for the problem of Product Recommendation in Online Advertising," Aug. 2018, [Online]. Available: http://arxiv.org/abs/1808.00720

[10] J. Feng *et al.*, “Learning to collaborate: Multi-scenario ranking via multi-agent reinforcement learning,” in *The Web Conference 2018 - Proceedings of the World Wide Web Conference, WWW 2018*, Apr. 2018, pp. 1939–1948. doi: 10.1145/3178876.3186165.

[11] X. Zhao, L. Xia, J. Tang, and D. Yin, “‘Deep reinforcement learning for search, recommendation, and online advertising: a survey’ by Xiangyu Zhao, Long Xia, Jiliang Tang, and Dawei Yin with Martin Vesely as coordinator,” *ACM SIGWEB Newsletter*, no. Spring, pp. 1–15, Jul. 2019, doi: 10.1145/3320496.3320500.

[12] Z. Dai, “Applying Reinforcement Learning Based Tutor Strategy Recommendation Service To The ASSISTments.”

[13] M. Langheinrich Ł, A. Nakamura, N. Abe, T. Kamba, and Y. Koseki, “Unintrusive customization techniques for Web advertising,” 1999.

[14] X. Zhao *et al.*, “DEAR: Deep Reinforcement Learning for Online Advertising Impression in Recommender Systems,” 2021. [Online]. Available: www.aaai.org

[15] C. Watkins and R. Holloway, “Learning From Delayed Rewards A reversible MCMC model of sexual evolution: practical genetic algorithms with closed-form stationary distributions View project.” [Online]. Available: https://www.researchgate.net/publication/33784417

[16] P. Auer, O. Cesa-bianchi, Y. Freund, R. E. Schapire, and S. J. Comput, “THE NONSTOCHASTIC MULTIARMED BANDIT PROBLEM *.” [Online]. Available: http://www.siam.org/journals/sicomp/32-1/39837.html

[17] A. Hojjat, J. Turner, S. Cetintas, and J. Yang, “A unified framework for the scheduling of guaranteed targeted display advertising under reach and frequency requirements,” *Oper Res*, vol. 65, no. 2, pp. 289–313, Mar. 2017, doi: 10.1287/opre.2016.1567.