Proceedings of the 4th International Conference on Signal Processing and Machine Learning
DOI: 10.54254/2755-2721/53/20241593

A survey on contextual multi-armed bandits

Qiufan Chen

University of Electronic Science and Technology of China, Chengdu, 611731, China

2669345625@qq.com

Abstract. As a powerful reinforcement learning framework, Contextual Multi-Armed Bandits have extensive applications in various domains. The models of Contextual Multi-Armed Bandits enable decision-makers to make intelligent choices in situations with uncertainty, and they find utility in fields such as online advertising, medical treatment optimization, resource allocation, and more. This paper reviews the evolution of algorithms for Contextual Multi-Armed Bandits, including traditional Bayesian approaches and the latest deep learning techniques. Successful case studies are summarized in different application domains, such as online ad click-through rate optimization and medical decision support. Furthermore, the author discusses future research directions, including more sophisticated context modeling, interpretability, fairness issues, and ethical considerations in the context of automated decision-making.

Keywords: Multi-arm bandit, Contextual bandit, Recommendation system.

1. Introduction

In recent years, contextual multi-armed bandits (CMBA) have been widely used in complex decision-making processes, where agents make a sequence of decisions from a set of arms whose reward depends on contextual information. The time in the decision-making process goes like 1, 2, ..., T, and the agent needs to make a decision to pull one arm among a set of K arms at each time T. Each arm has an unknown reward distribution that depends on the context, and the purpose of the agent is to learn which arm to select in order to maximize cumulative rewards over time. The reward of the arm being pulled will be received and that of the other arms remains unknown. In terms of the distribution of the reward, it is sampled from an unknown distribution in a stochastic setting while chosen by an adversary in an adversarial setting. It is worth noting that in various contexts, the arm getting the highest reward may also be different.

Applications of CMAB span various domains, from online advertising to healthcare treatment optimization [1]. By adapting strategies based on observed contexts and rewards, CMAB algorithms offer an intelligent approach to maximize cumulative rewards over time. As an overview, Table 1 summarizes all the algorithms discussed in this paper. In Table 1, the number of distinct contexts is represented by C, the number of policies is represented by N, the number of arms is represented by K, and the dimension of contexts is represented by d.

Through a review, this paper aims to inspire researchers' interest in Contextual Multi-Armed Bandits, fostering further exploration and innovation to address the evolving decision challenges in the real world.

DOI: 10.54254/2755-2721/53/20241593

Table 1. A comparison between all the algorithms of contextual bandits discussed in this paper.

Algorithm	Regret
LinUCB	$O(d\sqrt{T\ln((1+T)/\delta)})$
LinTS	$O(d^2\delta^{-1}\sqrt{T^{1+\delta}})$
D-LinUCB	$O((1/\epsilon^2+1/\Delta)\log(T))$
D-RandLinUCB	$\mathcal{O}(d^{\frac{2}{3}}\bar{B}_T^{\frac{1}{3}}T^{\frac{2}{3}})$
D-LinTS	$\mathcal{O}(d^{\frac{2}{3}}(\log K)^{\frac{1}{3}}B_T^{\frac{1}{3}}T^{\frac{2}{3}})$
TS for High-Dimensionl	$O(s^*\sqrt{T\log d})$ $O(s^{*2}[\log d+\log T]\log T)$

2. Contextual bandits feature representation

In the context of CMAB problems, an essential component is the process of feature extraction. CMAB problems involve an intelligent agent making decisions within various contextual scenarios to optimize cumulative rewards. The feature extraction step plays a pivotal role in mapping contextual information to appropriate action choices. In this regard, several established methods for feature extraction in CMAB scenarios are worth noting.

Firstly, the use of one-hot encoding is a prevalent technique [2]. It involves converting discrete contextual features, such as user demographics (e.g., gender or age groups), into binary vectors. Each unique feature value corresponds to a distinct dimension within the vector space. This method is particularly effective for scenarios with a finite set of discrete features.

Another approach is the application of embeddings [2]. Embeddings are advantageous when dealing with contextual features characterized by a multitude of potential values. They facilitate the transformation of such features into lower-dimensional continuous vector spaces, allowing for the capture of intricate relationships among features.

In addition, the normalization and standardization of continuous features are customary preprocessing steps [2]. These steps ensure that continuous features maintain consistent scales and ranges, which can be crucial for effective learning in CMAB settings.

Considering temporal dynamics, incorporating time-related features, such as timestamps or time intervals, can be essential. These features enable the model to account for temporal dependencies, which can be particularly relevant in scenarios where time plays a significant role.

Deep learning models, including Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs), offer a sophisticated approach to feature extraction. These models are adept at automatically deriving complex feature representations from contextual information.

Lastly, the combination of contextual features can yield more intricate representations. Techniques such as cross-feature interactions and polynomial feature engineering can be applied to enrich the feature set.

The choice of a specific feature extraction method within the CMAB framework depends on the problem's characteristics and the nature of the data at hand. It is imperative to experiment and fine-tune these methods to ascertain the most effective approach for maximizing rewards in CMAB scenarios.

3. Contextual bandits algorithms

3.1. Stochastic contextual bandits

Stochastic contextual bandit algorithms typically operate under the assumption that the reward associated with each arm conforms to an undisclosed probability distribution. Certain algorithms even

DOI: 10.54254/2755-2721/53/20241593

extend this assumption to posit that the distribution adheres to a sub-Gaussian distribution with unspecified parameters. Under the linear realizability assumption, this section discusses stochastic contextual bandit algorithms.

3.1.1. LinUCB. In the LinUCB algorithm [3], we assume that on each arm, the feature vector $x_{t,a} \in R^d$ is linear with the expected reward:

$$E[r_{t,a} \mid x_{t,a}] = x_{t,a}^{\top}\theta^* \tag{1}$$

Where θ^* is the true coefficient vector. Assume that at time t, the best arm is $a_t^* = \arg\max_a x_{t,a}^{\top}\theta^*$, then the regret of LinUCB after t round is defined as

$$\begin{aligned} R_T &= E\left[\sum_{t=1}^{T} r_{t,a_t^*} - \sum_{t=1}^{T} r_{t,a_t}\right] \\ &= \sum_{t=1}^{T} x_{t,a_t^*}^{\top}\theta^* - \sum_{t=1}^{T} x_{t,a_t}^{\top}\theta^* \end{aligned} \tag{2}$$

Let D_t represent the feature vector of the arm that is pulled at each time. Let c_t represent the corresponding reward. If the sample $(x_{t,a}, r_{t,a_t})$ is independent, then a closed-form estimator of θ^* is obtained through ridge regression

$$\hat{\theta}_t = (D_t^{\top}D_t + \lambda I_d)^{-1}D_t^{\top}c_t \tag{3}$$

For a prediction $x_{t,a}^{\top}\hat{\theta}_t$, the upper confidence bound should be:

Assume that the rewards $r_{t,a}$ are independent random variables with means $E[r_{t,a}] = x_{t,a}^{\top}\theta^*$, let $\epsilon = \sqrt{\frac{1}{2}\ln\frac{2TK}{\delta}}$, and $A_t = D_t^{\top}D_t + I_d$, then with the probability $1 - \delta/T$, we got

$$| x_{t,a}^{\top}\hat{\theta}_t - x_{t,a}^{\top}\theta^* | \leq (\epsilon + 1)\sqrt{x_{t,a}^{\top}A_t^{-1}x_{t,a}} \tag{4}$$

However, in the LinUCB algorithm, samples from the previous round are used to estimate θ^* and a sample for the present round is chosen, thus the samples are not independent.

3.1.2. LinTS. As an effective approach for balancing exploration and exploitation, Thompson sampling produces good empirical results with respect to display ads and news recommendations [4]. Multi-armed bandit problems can be solved using Thompson sampling as well. For contextual bandits, Agrawal et al. and Li et al. provide a Thompson sampling algorithm with linear payoffs [5,6]. Assume that in the bandit problem, there are K arms, and at time t, each arm "a" is linked to a d-dimensional feature vector $x_{t,a}$. The context selection is not assumed to follow a specific distribution and can be determined by an adversary. A d-dimensional parameter $\mu \in \mathbb{R}^d$ is used to define a linear predictor and predict the mean reward of arm a by $\mu \cdot x_{t,a}$. We assume an unknown underlying parameter $\mu^* \in \mathbb{R}^d$, therefore, at time t, the expected reward for the arm a is given by $\bar{r}_{t,a} = \mu^* \cdot x_{t,a}$. The actual reward $r_{t,a}$ is obtained from choosing the arm a at time t with an unknown distribution with mean $\bar{r}_{t,a}$. The Thompson sampling algorithm selects an arm a_t at each time $t \in \{1,...,T\}$ and receives a reward r_t. Let a^* be the optimal arm at time t:

$$a_t^* = \arg\max \bar{r}_{t,a} \tag{5}$$

Let $\Delta_{t,a}$ represent the difference of the expected reward between the optimal arm and arm a:

$$\Delta_{t,a} = \bar{r}_{t,a^*} - \bar{r}_{t,a} \tag{6}$$

Mathematically, the regret can be expressed as:

DOI: 10.54254/2755-2721/53/20241593

$$R_T = \sum_{t=1}^{T} \Delta_{t,a_t} \tag{7}$$

In the mentioned paper, the assumption

$\delta_{t,a} = r_{t,a} - \bar{r}_{t,a}$ is conditionally R-sub-Gaussian, which implies that for a constant R≥0, $r_{t,a} \in [\bar{r}_{t,a} - R, \bar{r}_{r,t} + R]$.

There are various likelihood distributions that satisfy this condition, but for the sake of simplicity, the paper assumes a Gaussian likelihood and Gaussian prior.

Thus, the likelihood of reward $\bar{r}_{t,a}$, given the context $x_{t,a}$, is modeled using the probability density function $\mathcal{N}(x_{t,a}^{T}\mu^{*}, v^2)$of a Gaussian distribution. Mathematically,v is defined as $v = R\sqrt{\frac{24}{\epsilon} d \ln(\frac{t}{\delta})}$, where the algorithm parameter is $\epsilon \in (0,1)$, and δ is a parameter that controls the high probability regret bound.

Similar to the closed-form of linear regression, we define

$$B_t = I_d + \sum_{\tau=1}^{t-1} x_{\tau,a} x_{\tau,a}^{\top} \tag{8}$$

$$\hat{\mu}_t = B_t^{-1}\left(\sum_{\tau=1}^{t-1} x_{\tau,a} r_{\tau,a}\right) \tag{9}$$

With probability 1−δ, the regret is bounded by:

$$R_T = O\left(\frac{d^2}{\epsilon}\sqrt{T^{1+\epsilon}}(\ln(Td)\ln\frac{1}{\delta})\right) \tag{10}$$

3.1.3. D-LinUCB. LinUCB is an optimistic algorithm for non-stationary environments in a stochastic linear bandit model [7]. To handle non-stationarity and forget past observations smoothly, it utilizes discounted linear regression and exponential weights $w_t = \gamma^{-t}$, where 0< γ<1 is the discount factor. The algorithm incorporates regularization and computes the upper confidence bound(UCB) for each action based on estimated regression parameters and uncertainty. The action with the highest UCB is selected to play, and the algorithm updates the parameter estimation and uncertainty based on observed rewards and chosen actions.

At each step, the algorithm receives a set of available actions A_t and computes an upper confidence bound(UCB) $UCB(a) = a^{\top}\hat{\theta} + \beta_{t-1}\sqrt{a^{\top}V^{-1}\tilde{V}V^{-1}a}$ for each action based on the current estimates of the unknown regression parameter $\hat{\theta}_t = \arg\min_{\theta\in\mathbb{R}^d}(\sum_{s=1}^{t} w_s(X_s - \langle A_s, \theta\rangle)^2 + \lambda_t \| \theta \|_2^2)$ and the uncertainty in the estimates. The algorithm selects the action with the highest UCB and plays it, receiving a reward $X_t = \langle A_t, \theta_t^{\star}\rangle + \eta_t$, The algorithm then updates its estimates of θ and the uncertainty in the estimates based on the played action and reward.

In the D-LinUCB algorithm, we introduce a confidence ellipsoid C_t which is defined as $\{\theta: \| \theta - \hat{\theta}_{t-1} \|_{V_{t-1}\tilde{V}_{t-1}^{-1}V_{t-1}} \leq \beta_{t-1}\}$, and let

$$\beta_t = \sqrt{\lambda}S + \sigma\sqrt{2\log(1/\delta) + d\log(1 + \frac{L^2(1-\gamma^{2t})}{\lambda d(1-\gamma^2)})} \tag{11}$$

Based on the remark above regarding to scale-invariance, we can easily conclude that at time t, our D-LinUCB algorithm chooses the action A_t that maximizes $\langle a, \theta\rangle$ for $a \in \mathcal{A}_t$ and $\theta \in C_t$.

Assuming that $\sum_{s=1}^{T-1} \| \theta_s^{\star} - \theta_{s+1}^{\star} \|_2 \leq B_T$, the regret of the D-LinUCB algorithm is bounded for all $\gamma \in (0,1)$ and $D \geq 1$ then with the probability $1 - \delta/T$, we got

$$R_T \leq 2LDB_T + \frac{4L^3S}{\lambda}\frac{\gamma^D}{1-\gamma}T + 2\sqrt{2}\beta_T\sqrt{dT}\sqrt{T\log(1/\gamma) + \log(1 + \frac{L^2}{d\lambda(1-\gamma)})} \tag{12}$$

3.1.4. D-RandLinUCB. D-LinUCB algorithm follows the optimism in the face of the uncertainty principle and picks actions through maximizing the UCB bound of expected reward based on

DOI: 10.54254/2755-2721/53/20241593

confidence level a and $\hat{\theta}_t^{wls}$ [8]. However, in a non-stationary linear bandit environment, the D-RandLinUCB algorithm replaces the confidence level a with a random variable $Z_t \sim \mathcal{D}$.

D-LinUCB: $X_t = \arg\max_{x \in \mathcal{X}_t} \langle x, \hat{\theta}_t^{wls} \rangle + a \parallel x \parallel_{V_t^{-1}}$

D-RandLinUCB: $X_t = \arg\max_{x \in \mathcal{X}_t} \langle x, \hat{\theta}_t^{wls} \rangle + Z_t \parallel x \parallel_{V_t^{-1}}$.

In the non-stationary linear bandit environment, dynamic regret is introduced to quantify the cumulative regret incurred by an algorithm over a sequence of time steps, which allow us to evaluate the performance of various algorithm in non-stationary bandit settings.

Suppose that at time t, algorithm A chooses arm $X_t = \arg\max_{\mathcal{X}_t} \tilde{f}_t(x)$. The corresponding expected dynamic regret is bounded for integer D>0,

$$\begin{aligned} E[R(T)] &\leq (c_1 + c_2)\left(1 + \frac{2}{p_3 - p_2}\right)\sqrt{c_3 T} \\ &\quad + T(p_1 + p_2) + d + 2DB_T + \frac{4}{\lambda}\frac{\gamma^D}{1-\gamma}T. \end{aligned} \tag{13}$$

$$c_1 = \sqrt{2\log T + d\log\left(1 + \frac{1-\gamma^{2(T-1)}}{\lambda d(1-\gamma^2)}\right)} + \lambda^{1/2}, \tag{14}$$

$$\&c_2 = a\sqrt{2\log\left(\frac{T}{2}\right)}, \text{and } a^2 = 14c_1^2 \tag{15}$$

If we choose $D = \frac{\log T}{1-\gamma}, \gamma = 1 - (B_T/(dT))^{2/3}$, the expected dynamic regret is asymptotically upper bounded by $\mathcal{O}(d^{2/3}B_T^{1/3}T^{2/3})$ as $T \to \infty$.

In the D-RandLinUCB algorithm which is designed to overcome conservatism issues faced by optimism-based algorithms in practice, we use the weighted method with exponentially discounting factor to adjust the non-stationary linear bandit environment. Since the action set $\mathcal{X}_t$ changes from time t and has infinite arms, and the true parameter $\theta_t^\star$ varies within total variation B_T.

D-RandLinUCB achieves statistical optimality in terms of dynamic regret, but it has a trade-off with computational efficiency compared to another algorithm called Discounted Linear Thompson Sampling (D-LinTS).

3.1.5. D-LinTS. In the previous D-LinUCB algorithm [7], the random perturbations were injected by replacing optimism with simple randomization when deciding the confidence level. However, the D-LinTS algorithm chooses to perturb estimates before the expected rewards are maximized [8]. We use a weighted least-squares estimator $\hat{\theta}_t^{wls}$ along with the corresponding matrix $V_t = W_{t,\lambda}\widetilde{W}_{t,\lambda}^{-1}W_{t,\lambda}$ instead of $\hat{\theta}_t^{ls}$ and the Gram matrix $V_{t,\lambda}$. Since the random perturbations are not shared on each arm, the D-LinTS algorithm has more variation and corresponding larger regret bounds than the previous D-RandLinUCB algorithm.

In the D-LinTS algorithm [8], we choose $D = \log T/(1-\gamma)$ and $\gamma = 1 - (B_T/(dT\sqrt{\log K}))^{2/3}$, the expected dynamic regret is asymptotically upper bounded by $\mathcal{O}(d^{2/3}(\log K)^{1/3}B_T^{1/3}T^{2/3})$ as $T \to \infty$.

To ensure that the randomly chosen confidence bound of D-RandLinUCB belongs to that of D-LinUCB with high probability, D-RandLinUCB uses a truncated normal distribution with zero mean and standard deviation 2/5 over [0,∞) as D. On the other hand, when implementing both LinTS and D-LinTS, D-LinTS uses a non-inflated version by setting a=1. According to Li et al. [3], in all scenarios, both randomized algorithms outperform the non-randomized D-LinUCB. However, D-LinTS not only outperforms D-RandLinUCB in all scenarios but also works as well as Oracle Restart LinTS considering the high dimension and big action space.

DOI: 10.54254/2755-2721/53/20241593

3.2. High-dimensionl contextual bandits

In the domain of high-dimensional contextual bandits, we move beyond the traditional constraints of fixed probability distributions for arm rewards. Here, the arms are not bound by predefined distributions and can be strategically selected by an adversary, challenging the decision-making process. To navigate this complex landscape, advanced techniques tailored to high-dimensional contexts are employed. These methods empower agents to leverage the rich contextual information, often encoded in high-dimensional feature spaces, to make informed arm selections. The agent's strategy adapts dynamically based on observed rewards, allowing for intricate adjustments in the high-dimensional space to optimize decision-making in this adversarial environment.

The Thompson sampling algorithm is a popular Bayesian algorithm for solving the contextual bandit problem. In the high-dimensional and sparse contextual bandit problem, the algorithm uses special classes of sparsity-inducing priors [9], such as spike-and-slab priors, to model the unknown parameter. The algorithm works by first initializing the prior distribution over the unknown parameter, and then at each round, it samples a parameter from the posterior distribution, which is updated based on the observed rewards and contexts. The algorithm then selects the action that maximizes the expected reward under the sampled parameter. By using sparsity-inducing priors, the algorithm can effectively handle high-dimensional and sparse contexts, and by using Bayesian inference, it can provide a probabilistic estimate of the unknown parameter.

The algorithm used the sparsity-inducing prior proposed in the research of Russac et al. [10] for posterior sampling and established posterior contraction results for non-i.i.d. observations coming from a bandit environment and for a wide class of noise distributions. Using the posterior contraction result, an almost dimension-free regret bound is established for the proposed TS algorithm under different arm-separation regimes parameterized by ω. The algorithm enjoys minimax optimal performance for $\omega \in [0,1)$. In addition, the prior allows us to design a computationally efficient TS algorithm based on Variational Bayes.

First, a dimension s is selected from a prior π_d on the set [d]; next, a random subset $S \subset [d]$ of size $|S|=s$, and finally, given S, a set of nonzero values $\beta_S := \{\beta_i : i \in S\}$ from a prior density g_S for $\mathbb{R}^S$. In the tth round of the algorithm, a specific prior Π is set on β, and it is updated sequentially based on the observed rewards and contexts. In particular, it chooses the prior described in

$$(S, \beta) \mapsto \pi_d(|S|) \frac{1}{\binom{d}{|S|}} g_S(\beta_S)\delta_0(\beta_{S^c}) \tag{16}$$

with an appropriate choice of round-specific prior scaling λ_t and updates the posterior using the observed rewards and contexts until (t−1)th round. Then a sample is generated from the posterior and an arm a_t is chosen greedily based on the generated sample.

Since $C = \Theta(\phi_u \vartheta^2 \xi K \log K)$, and $K \geq 2, d \geq T$. Define the quantity

$$\kappa(\xi, \vartheta, K) := \min\{(4c_3 K \xi \vartheta^2)^{-1}, 1/2\} \tag{17}$$

Rewards where c3 is a universal positive constant. Also, set the prior scaling λ_t as follows:

$$(5/3)\bar{\lambda}_t \leq \lambda_t \leq 2\bar{\lambda}_t, \bar{\lambda}_t = x_{max}\sqrt{2t(\log d + \log t)} \tag{18}$$

Then there exists a universal constant C0>0 such that we have the following regret bound for the algorithm:

$$\mathbb{E}\{R(T)\} \lesssim I_b + I_\omega \tag{19}$$

where,

$$I_b = \left\{\frac{b_{max} x_{max} \phi_u \vartheta^2 \xi (K \log K)}{\min\{\kappa^2(\xi,\vartheta,K), \log K\}}\right\} s^* \log (Kd) \tag{20}$$

DOI: 10.54254/2755-2721/53/20241593

$$I_{\omega}=\begin{cases}\Phi^{1+\omega}\left(\frac{s^{*1+\omega}(\log d)^{\frac{1+\omega}{2}}T^{\frac{1-\omega}{2}}}{\Delta_{*}^{\omega}}\right), & for\omega\in[0,1),\\ \Phi^{2}\left(\frac{s^{*2}[\log d+\log T]\log T}{\Delta_{*}}\right), & for\omega=1,\\ \frac{\Phi^{2}}{(\omega-1)}\left(\frac{s^{*2}[\log d+\log T]}{\Delta_{*}}\right), & for\omega\in(1,\infty)\\ \Phi^{2}\left(\frac{s^{*2}[\log d+\log T]}{\Delta_{*}}\right), & for\omega=\infty,\end{cases} \tag{21}$$

and

$$\Phi=\sigma x_{max}^{2}\xi K\left(2+40A_{4}^{-1}+C_{0}K\xi x_{max}^{2}A_{4}^{-1}\right) \tag{22}$$

4. Practical applications

The nature of contextual bandit problems makes them suitable for various real-life situation and application (see Table 2). In particular, they can be beneficial when collecting data for assessing treatment effectiveness on animal models throughout different disease stages. Traditionally, conducting such assessments using conventional random treatment allocation procedures can be challenging. This is because administering poor treatments can lead to a deterioration of the subject's health, making data collection difficult. To address this issue, Durand et al. [11] intend to develop an adaptive allocation strategy that allocates more samples, so as to enhance the data-collection efficiency and explore promising treatments. In their work, the authors approach this application as a contextual bandit problem and introduce a practical algorithm for exploration and exploitation within this framework.

Table 2. Bandit for Real Life Application [12].

	CMAB	Non-stationary CMAB
Healthcare	√	
Recommendation system	√	√
Dialogue system	√	

In addition to real life applications like clinical trials, contextual bandits could also be used to improve various machine learning algorithms (see Table 3).

Bouneffouf et al. [13] explore this idea by introducing a novel active learning strategy that models the active learning problem as a contextual bandit problem. Their proposed method, called Active Thompson Sampling (ATS), adopts a sequential algorithmic approach. In each round of the algorithm, ATS assigns a sampling distribution on a pool of available unlabeled data points. From this distribution, it samples one point and queries the oracle for the corresponding label of the sampled point. This active learning strategy effectively balances exploration and exploitation by making informed decisions on which data points to query for label information.

Noothigattu et al. [14] conducted a study that focuses on a scenario where an agent can observe the behavioral traces of individuals within a society but lacks access to explicit constraints governing the observed behaviors. To address this challenge, the authors employ inverse reinforcement learning to learn these potential constraints. Once the constraints are learned, they are combined with a potentially unrelated value function using a contextual bandit-based orchestrator. This orchestrator plays a critical role in selecting a contextually-appropriate choice between two policies: the constraint-based policy and the environment reward-based policy. When making decisions, the agent can now mix policies in new approaches, selecting the best actions from either a reward-maximizing policy or a constrained policy.

DOI: 10.54254/2755-2721/53/20241593

Table 3. Bandit in Machine Learning [12].

	CMAB	Non-stationary CMAB
Active Learning	√	
Reinforcement learning	√	

5. Conclusion

This paper reviews some of the most notable algorithms of contextual multi-armed bandits and summarizes it, in an organized way (Table 1). The LinUCB algorithm is a linear bandit algorithm that balances exploration and exploitation by modeling the uncertainty of each arm's reward using a linear regression approach [3]. The LinTS algorithm is a linear bandit algorithm that uses Bayesian methods to estimate the uncertainty of each arm, achieving a balance between exploration and exploitation [5]. The D-LinUCB algorithm is an enhanced linear upper confidence bound algorithm that improves the accuracy of uncertainty estimation for each arm by introducing covariance information, achieving a better balance between exploration and exploitation [6]. The D-RandLinUCB algorithm combines randomization and LinUCB approach to achieve improved exploration-exploitation trade-off by effectively estimating the arm rewards and uncertainties [7]. Similarly, the D-LinTS algorithm enhances the LinTS approach by incorporating randomization, leading to more effective uncertainty modeling and a better balance between exploration and exploitation in linear bandit problems [7]. In a high-dimensional environment, the TS algorithm [8] usually employs Bayesian methods to estimate uncertainty for multi-dimensional arms, achieving a more precise balance between exploration and exploitation.

In conclusion, this review has highlighted key insights and trends in contextual multi-armed bandits. It is evident that the traditional UCB algorithm and TS algorithm are no longer sufficient to address the decision-making challenges posed by complex environments in multi-armed bandit problems and that this field has seen significant developments in recent years. While we have discussed important contributions, it's essential to acknowledge the existing uncertainties and the need for further research. Looking ahead, future research in this area could explore extension in multimodal environments or integration of deep learning and reinforcement learning, and the findings presented here are expected to have a lasting impact on many Recommendation Systems like medical decision-making and automated trading.

References

[1] Bietti, A., Agarwal, A. and Langford, J. (2021). A contextual bandit bake-off, arXiv.org. Available at: https://arxiv.org/abs/1802.04064 (Accessed: 08 September 2023).

[2] Lin, B. et al. (2020). Contextual bandit with adaptive feature extraction, arXiv.org. Available at: https://arxiv.org/abs/1802.00981 (Accessed: 08 September 2023).

[3] Li, L. et al. (2010). A contextual-bandit approach to personalized news article recommendation. Proceedings of the 19th international conference on World wide web [Preprint]. doi:10.1145/1772690.1772758.

[4] Chapelle, O. and Li, L. (2011). An empirical evaluation of thompson sampling. Advances in neural information processing systems, 24.

[5] Agrawal, S. and Goyal, N. (2014). Thompson sampling for contextual bandits with linear payoffs, arXiv.org. Available at: https://arxiv.org/abs/1209.3352 (Accessed: 08 September 2023).

[6] Li, L., Chu, W., Langford, J. and Schapire, R. E. (2010). A Contextual-Bandit Approach to Personalized News Article Recommendation. arXiv.org. https://doi.org/10.1145/1772690.1772758.

DOI: 10.54254/2755-2721/53/20241593

[7] Russac, Y., Vernade, C. and Cappé, O. (2020). Weighted Linear Bandits for non-stationary environments, arXiv.org. Available at: https://arxiv.org/abs/1909.09146 (Accessed: 08 September 2023).

[8] Kim, B. and Tewari, A. (2021). Randomized exploration for non-stationary stochastic linear bandits, arXiv.org. Available at: https://arxiv.org/abs/1912.05695 (Accessed: 08 September 2023).

[9] Chakraborty, S., Roy, S. and Tewari, A. (2023). Thompson sampling for high-dimensional sparse linear contextual bandits, PMLR. Available at: https://proceedings.mlr.press/v202/chakraborty23b.html (Accessed: 08 September 2023).

[10] Castillo, I., Schmidt-Hieber, J., and Van der Vaart, A. (2015). Bayesian linear regression with sparse priors. The Annals of Statistics, 43(5), 1986–2018.

[11] Durand, A., Achilleos, C., Lacovides, D., Strati, K., Mitsis, G. D. and Pineau, J. (2018). Contextual bandits for adapting treatment in a mouse model of de novo carcinogenesis. In Machine Learning for Healthcare Conference, 67–82.

[12] Bouneffouf, D. and Rish, I. (2019). A survey on practical applications of multi-armed and Contextual Bandits, arXiv.org. Available at: https://arxiv.org/abs/1904.10040 (Accessed: 08 September 2023).

[13] Bouneffouf, D., Laroche, R., Urvoy, T., Feraud, R. and Allesiardo, R. (2014). Contextual bandit for active learning: Active thompson sampling. In International Conference on Neural Information Processing, 405–412. Springer.

[14] Noothigattu, R., Bouneffouf, D., Mattei, N., Chandra, R., Madan, P., Varshney, K., Campbell, M., Singh, M. and Rossi, F. (2018). Interpretable multi-objective reinforcement learning through policy orchestration. arXiv preprint arXiv:1809.08343.