ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) | 978-1-7281-6327-7/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICASSP49357.2023.10095479

QUESTION ANSWERING SYSTEM WITH SPARSE AND NOISY FEEDBACK

Djallel Bouneffouf[1], *Oznur Alkan*[2], *Raphael Feraud*[3], *Baihan Lin*[4]

[1] IBM Thomas J. Watson Research Center, Yorktown Heights, NY USA
[2] United Health Group [3] Orange Labs
[4] Columbia University

ABSTRACT

The rise of personal assistants has made question answering a very popular mechanism for user-system interaction. In Question Answering System, implicit feedbacks can be easily observed (user clicking in the link given by the QA system), but they are noisy. However, receiving an explicit feedback on the quality of the response just given is rare but more valuable. Motivated by a practical need in Question Answering System of processing these two types of rewards, this paper investigates and proposes a new stochastic multi-armed bandit model in which each action has a noisy reward and a sparse reward. We studied this problem in the contextual bandit settings, and proposed and analyzed efficient algorithms that are based on the LINUCB frameworks. Our algorithms are verified by empirical studies on various reward distributions and a real-world dataset and application.

Index Terms— QA System, Online Learning, Bandit, Contextual Bandit.

1. INTRODUCTION

Sequential decision making is a common problem in many practical applications where the agent must choose the best action to perform at each iteration in order to maximize the cumulative reward over some period of time. One of the key challenges in this process is to achieve a good trade-off between the exploration of new actions and the exploitation of known actions. This exploration vs exploitation trade-off in sequential decision making is often formulated as a *multi-armed bandit (MAB)* problem: given a set of bandit "arms" (actions), each associated with a fixed but unknown reward probability distribution [1, 2, 3, 4, 5, 6], an agent selects an arm to play at each iteration, and receives a reward drawn according to the selected arm's distribution independently from the previous actions.

A particularly useful version of MAB is the *contextual multi-armed bandit (Contextual-MAB)*, or simply the *contextual bandit* problem, where at each iteration, before choosing an arm, the agent observes an N-dimensional *context*, or *feature vector*. Over time, the goal is to learn the relationship between the context vectors and the rewards in order to make better predictions of which action to choose given the context [7]. In this paper, we consider a new problem setting, referred to as *bandit with Sparse and Noisy rewards*, where the agent observes two rewards, one which is noisy and another one which is sparse.

This setting is motivated by a real-world applications. For Question answering, the online services sequentially choose a response among several alternatives and display this option to the user. Customers clicking on the displayed option can indicate their interest, however a less noisy feedback could be observed by having the user rating the the QA system answer. In this specific setting, the user clicking on a chosen content can be considered as a noisy reward which is a proxy for the sparse reward. Another scenario of receiving noisy rewards in addition to sparse rewards can occur in movie recommendation settings. When the user chooses to watch a movie, this can be seen as a positive feedback on the movie. Then, the sparse reward is the rating that the user may give or not.

Motivated by the above scenarios, this paper focuses on handling the problem of multi-armed bandit with Sparse and Noisy rewards. The bandit framework proposed here aims to capture the scenarios described above through providing an approach so as to always exploit the noisy rewards. We first review some existing methods in multi-armed bandit and propose extensions to accommodate our problem setup. Then we proceed by proposing novel algorithm called ILINUCB, for this setting. Finally, we demonstrate the effectiveness of the proposed algorithms with experiments on various reward distributions and a real-world dataset.

2. RELATED WORK

The multi-armed bandits provide a solution to the exploration versus exploitation trade-off, informing a player how to pick an action within a finite set of decisions while maximizing cumulative reward in an online learning setting. Optimal solutions have been developed for a variety of problem formulations [2, 8, 9, 10, 11]. In Linear Upper Confidence Bound (LINUCB) [12, 13] and in Contextual Thompson Sampling (CTS) [7, 14], the authors assume a linear dependency between the expected reward of an action and its context, where

the representation space was modeled using a set of linear predictors.

Several authors have considered different types of rewards. A variant of the stochastic MAB problem is discussed in [15], where the rewards are corrupted. In this framework, motivated by privacy preserving in online recommender systems, the goal is to maximize the sum of the (unobserved) rewards, based on the observation of transformation of these rewards through a stochastic corruption process with known parameters. Another variation on the type of reward is explored in [16], authors present a general framework to model the relationship between partial and delayed feedback for the best arm identification problem in multi-armed bandits. Similarly, in [17], the authors study reinforcement learning with two streams of rewards where one reward is positive and the other one is negative.

A notion of more than one reward at a time is studied in [18], which is motivated by an online advertisement system. In order to reduce the cost of communication, the ad exchange stores the reward, and sends a pile of rewards to the system when sufficient number of rewards are gathered. The authors adapt the LINUCB algorithm to deal with the piled-reward setting. Similarly, and motivated by delayed conversions in advertising, [19, 20, 21] authors present a delay-adaptive algorithm for generalized linear contextual bandits using UCB-style exploration. Note that, this delayed reward setting could be another direction to explore the two rewards setting that we are proposing in this paper, where one of the rewards is delayed. These algorithms assume that the bandit can observe one reward at each iteration, which is different than our problem setting, where the learner receives two rewards at each iteration such that, one of the rewards is more noisy than the other.

In [22], multiple rewards at each iteration is studied where the goal is the multi-objective aggregation of rewards. They presented a novel approach for dynamically optimizing multiple reward metrics simultaneously via multi-armed bandit approach in the context of language generation. Note that, this work is dealing with rewards for different objective, where in our case we have one objective. Compared to the previous related art, to the best of our knowledge, this is the first work to study the problem of consuming two types of rewards after each action.

3. CONTEXTUAL BANDIT WITH SPARSE AND NOISY FEEDBACK.

In this setting (Algorithm 1), at each time $t \in [T]$, a player is presented with a *context vector* $\mathbf{x}_t \in \mathbb{R}^d$, where $\|\mathbf{x}_t\|_2 \leq 1$, and must choose an arm $k \in [K]$. We operate under the linear realizability assumption, i.e., for all $k \in [K]$, there exist unknown weight vectors $\boldsymbol{\theta}_k \in \mathbb{R}^d$ with $\|\boldsymbol{\theta}_k\|_2 \leq 1$ so that $\forall t$:

$$\mathbb{E}[r_k(t)|\mathbf{x}_t] = {\boldsymbol{\theta}_k}^\top \mathbf{x}_t = {\boldsymbol{\theta}_k^n}^\top \mathbf{x}_t + L{\boldsymbol{\theta}_k^s}^\top \mathbf{x}_t,$$

where $\boldsymbol{\theta}_k^n$ and $\boldsymbol{\theta}_k^s$ are respectively the optimal parameters for the noisy reward r_k^n and the sparse reward r_k^n.

Algorithm 1 Contextual Bandit with Sparse and Noisy Feedback

1: **Repeat**
2: $(\mathbf{x}_t, \mathbf{r}_t)$ is drawn according to some distribution
3: $\mathbf{x}_t$ is revealed to the player
4: The player chooses an action k
5: The reward $r_k^n(t)$ is revealed
6: The reward $r_k^s(t)$ is revealed with a probability L
7: The player updates its parameter $\hat{\boldsymbol{\theta}}_k(t)$
8: $t \leftarrow t + 1$
9: **Until** t=T

Assumption 1. *The two rewards are independent, but we assume a fixed and known gap between their expectations:*

$$\forall k \in [K], \forall t, \quad {\boldsymbol{\theta}_k^s}^\top \mathbf{x}_t = {\boldsymbol{\theta}_k^n}^\top \mathbf{x}_t + \phi, \text{ where } \phi \in \mathbb{R}. \quad (1)$$

Assumption 2 (Sub-Gaussian noise:)**.** *$\forall\, x \in X$ and $\forall k \in [K]$, the noise of the sparse $\epsilon_k^s(t) = r_k^s(t) - \mathbf{x}_t^\top \boldsymbol{\theta}_k$ and the noise of the noisy rewards $\epsilon_k^n(t) = r_k^n(t) - \mathbf{x}_t^\top \boldsymbol{\theta}_k$ are conditionally ρ_s-sub-Gaussian and ρ_n-sub-Gaussian respectively, with $\rho_n \geq \rho_s \geq 0$, that is for all $t \geq 1$,*

$$\forall\, \lambda \in \mathbb{R}, \quad E[e^{\lambda \epsilon_k^i(t)}] \leq \exp\left(\frac{\lambda^2 \rho_i^2}{2}\right).$$

with $i \in \{n, s\}$

Definition 1 (Cumulative regret in our setting)**.** *The pseudo-regret at time T is given as:*

$$\begin{aligned} R(T) &= \sum_{t=1}^{T} {\boldsymbol{\theta}_k}^\top \mathbf{x}_t - \sum_{t=1}^{T} \hat{\boldsymbol{\theta}}_k(t)^\top \mathbf{x_t}, \\ &= \sum_{t=1}^{T} \left[{\boldsymbol{\theta}_k^n}^\top \mathbf{x}_t + L{\boldsymbol{\theta}_k^s}^\top \mathbf{x}_t\right] - \sum_{t=1}^{T} \left[\hat{\boldsymbol{\theta}}_k^n(t)^\top \mathbf{x}_t + L\hat{\boldsymbol{\theta}}_k^s(t)^\top \mathbf{x}_t\right], \\ &= (1+L) \sum_{t=1}^{T} \left[{\boldsymbol{\theta}_k^n}^\top \mathbf{x}_t - \hat{\boldsymbol{\theta}}_k^n(t)^\top \mathbf{x}_t\right]. \end{aligned} \quad (2)$$

3.1. Algorithm description

One solution to the contextual bandit problem is the LINUCB algorithm proposed in [12], where the key idea is to apply online ridge regression to incoming data to obtain an estimate of the coefficients $\boldsymbol{\theta}_k$ for $k = 1, \ldots, K$. At each time step t, the LINUCB policy selects the arm with the highest upper confidence bound of the reward $k(t) = \text{argmax}_k(\boldsymbol{\theta}_k^\top \mathbf{x}_t + c_k)$, where $c_k = \alpha\sqrt{\mathbf{x}_t^\top \mathbf{A}_k^{-1} \mathbf{x}_t}$ is the standard deviation of the corresponding reward scaled by exploration-exploitation trade-of parameter α (chosen a priori) and $\mathbf{A}_k$ is the covariance of the k-th arm context. LINUCB requires a reward for

the chosen arm k, $r_k(t)$, to be observed to perform its updates. In our setting, since the learner received two rewards, the noisy and the sparse rewards $r_k^n \in \mathbb{R}$ and $r_k^s \in \mathbb{R}$, we need to adjust LINUCB algorithm to learn from the noisy rewards as well.

In order to be robust to variations in the noisy reward's quality, we only allow noisy reward to vary within agent's beliefs. To make use of the noisy rewards, we consider a filtering mechanism that constrain the noisy reward to be within the upper and lower bound of the expected sparse reward for the chosen arm: where $r_k^{n'}(t)$ denotes the filtered reward. The intuition here is to use the upper and lower bound of the sparse reward as an index of the quality of the noisy rewards. If the noisy reward is in the confidence interval of the sparse reward c_k^s (see Algorithm 2, line 11), we accept it, otherwise we replace it by the lower or the upper bound of the sparse reward.

$$\begin{cases} r_k^n(t) \geq \hat{\boldsymbol{\theta}}_k^{s\top}\mathbf{x}_t - \phi + c_k^s, & r_k^{n'}(t) \leftarrow \hat{\boldsymbol{\theta}}_k^{s\top}\mathbf{x}_t - \phi + c_k^s, \\ r_k^n(t) \leq \hat{\boldsymbol{\theta}}_k^{s\top}\mathbf{x}_t - \phi - c_k^s, & r_k^{n'}(t) \leftarrow \hat{\boldsymbol{\theta}}_k^{s\top}\mathbf{x}_t - \phi - c_k^s, \\ \textbf{else } r_k^{n'}(t) \leftarrow r_k^n(t). \end{cases} \tag{3}$$

Equation 3 in the proposed algorithm (ILINUCB) only allows noisy rewards to vary within agent's beliefs. ILINUCB is summarized in Algorithm 2.

Algorithm 2 ILINUCB for contextual Multi-armed bandit

1: **Input:** α
2: $\forall k \in [K], \boldsymbol{A}_k \leftarrow I_{d+1}, \boldsymbol{S}_k \leftarrow I_{d+1}, \boldsymbol{b}_k \leftarrow \mathbf{0}_{d+1}, \boldsymbol{b}_k^s \leftarrow \mathbf{0}_{d+1}, \boldsymbol{\theta}_k \leftarrow \mathbf{0}_{d+1}, \boldsymbol{\theta}_k^s \leftarrow \mathbf{0}_{d+1}$.
3: **for** $t = T_0 + 1$ **to** T **do**
4: observe x_t
5: **for** all $k \in [K]$ **do**
6: $\hat{\boldsymbol{\theta}}_k \leftarrow \mathbf{A}_k^{-1} * \boldsymbol{b}_k, c_k \leftarrow \alpha\sqrt{\mathbf{x}_t^\top \mathbf{A}_k^{-1}\mathbf{x}_t}$,
7: $\hat{\boldsymbol{\theta}}_k^s \leftarrow \mathbf{S}_k^{-1} * \boldsymbol{b}_k^s, c_k^s \leftarrow \alpha\sqrt{\mathbf{x}_t^\top \mathbf{S}_k^{-1}\mathbf{x}_t}$
8: **end for**
9: play arm $k = \arg\max_k(\hat{\boldsymbol{\theta}}_k^\top \mathbf{x}_t + c_k)$
10: observe $r_k^n(t)$ and if $r_k^s(t)$ exist then $h(t) \leftarrow 1$ else $h(t) \leftarrow 0$
11: Compute $r_k^{n'}$ according to equation (3)
12: $r_k'(t) \leftarrow h(t) r_k^s(t) + r_k^{n'}(t)$
13: $\mathbf{S}_k \leftarrow \mathbf{S}_k + h(t)\mathbf{x}_t\mathbf{x}_t^\top, \boldsymbol{b}_k^s \leftarrow \boldsymbol{b}_k^s + h(t) r_k^s(t)\mathbf{x}_t$
14: $\mathbf{A}_k \leftarrow \mathbf{A}_k + \mathbf{x}_t\mathbf{x}_t^\top$,
15: $\boldsymbol{b}_k \leftarrow \boldsymbol{b}_k + r_k'(t)\mathbf{x}_t$
16: **end for**

4. EXPERIMENTAL RESULTS

We compared the performance of our proposed algorithms in various experimental settings. In these experiments, our aim was to investigate the effect of the sparsity and the level of noise in the noisy feedback on the performance of the algorithms. Since the problem is new, there is no directly comparable solution in existing work that considers noisy and sparse rewards. Therefore, we compared our proposed algorithm ILinUCB with D-LinUCB which only consumes sparse rewards and ID-LinUCB which consumes noisy rewards but in an uncontrolled manner such that it does not perform the bound check as outlined in our algorithm.

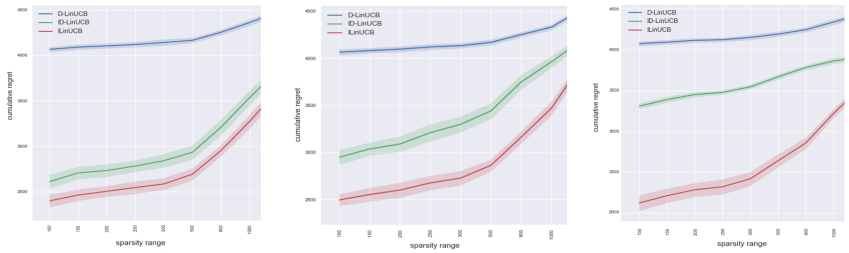
We experimented on two parameters: **L**:sparsity, **p**: probability of noise in noisy rewards. For L, we experimented with different sparsity levels. Going from one observed reward by 100 iterations to one observed reward by 1000 iterations.

For p, we experimented with values within the range [0-1], where *p=0* represents the case of having perfect noisy rewards, in other words, the noisy rewards being exactly the same as the sparse rewards. *p=1* represents the setting where noisy rewards are always noisy. Therefore, as p increases, the noise in the noisy reward increases. For each test case, we run the experiments for 100 times, and take the average of both the correctness metrics and the number of pulls. note that in all experiments we are setting the $\Phi = 0$.

For evaluating ILinUCB, Warfarin dataset [23] is used. This data set is concerned with determining the correct initial dosage of the drug Warfarin for a given patient and it contains 5528 patients' records where each record has 65 features. The dosage is discretized to three levels: low, medium and high. In our setting, each record corresponds to a timepoint and each dosage level corresponds to an arm. For each time point, if the correct arm is pulled, then the learner receives a sparse reward of value 1, otherwise 0.

Selected results of the experiments with ILinUCB and the baselines are presented in Figure 1. For lower values of p, we can observe the benefit of consuming noisy rewards as both ID-LinUCB and ILinUCB have lower regret than D-LinUCB. This is as expected since if we have additional feedback that are good approximations of sparse feedback, this results in less total regret. When the noise in the feedback increases, the gap between the regret of ILinUCB and ID-LinUCB increase. This proves the benefit of controlling the consumption of noisy feedback with the bound check in ILinUCB. This finding confirms with the findings we observed from the experiments with non contextual settings in terms of the effect of sparsity and noise in rewards on the overall performance of our proposed algorithms.

We can conclude two important findings from these experiments. First, if we have additional evidence which is to a degree well aligned with the sparse rewards, we can have lower regrets through consuming these noisy additional evidences. However, if there is a possibility of high noise in the noisy rewards, they should be bounded by the algorithm as otherwise the noise can increase the regret of the algorithm significantly.



(a) $p = 0.2$, (b) $p = 0.3$, (c) $p = 0.5$

Fig. 1. Experiments with ILinUCB and baselines for varying sparsities and probabilities of noise in rewards on Warfarin dataset.

4.1. Customer Assistant Evaluation:

Next we evaluate our methods on *Customer Assistant*, a proprietary multi-skill dialog orchestration dataset. Recall that this kind of application motivates the contextual bandit with sparse and noisy feedback setting because there is a naturally two types of rewards that we can collect:

1- Noisy rewards: this reward is an estimated reward that we get from the time the user spent in the clicked link recommended. So we basically use the historical data to define the maximum and minimum time spent in a clicked link, and normalize the reward to be between [0,1].

2- Sparse rewards: this rewards based on the number of stars the users is giving to the answer he received, so we have a five stars evaluation maximum and 0 minimum, and we normalize the results to be between [0,1].

The Customer Assistant orchestrates 9 domain specific agents which we arbitrarily denote as $Skill_1, \ldots, Skill_9$ in the discussion that follows. In this application, example skills lie in the domains of payroll, compensation, travel, health benefits, and so on. In addition to a textual response to a user query, the skills orchestrated by *Customer Assistant* also return the following features: an *intent*, a short string descriptor that categorizes the perceived intent of the query, and a *confidence*, a real value between 0 and 1 indicating how confident a skill is that its response is relevant to the query. Skills have multiple intents associated with them. The orchestrator uses all the features associated with the query and the candidate responses from all the skills to choose which skill should carry the conversation.

The Customer Assistant dataset contains 28,412 events associated with a correct skill response. We encode each query by averaging 50 dimensional GloVe word embeddings for each word in each query and for each skill we create a feature set consisting of its confidence and a one-hot encoding of its intent. The skill feature set size for $Skill_1, \ldots, Skill_9$ are 181, 9, 4, 7, 6, 27, 110, 297, and 30 respectively. We concatenate the query features and all of the skill features to form a 721 dimensional context feature vector for each event in this dataset.

In a live setting the query features are immediately calculable or known, whereas the confidence and intent necessary to build a skill's feature set are unknown until a skill is executed. Because the confidence and intent for a skill are both accessible post execution, we reveal them together.

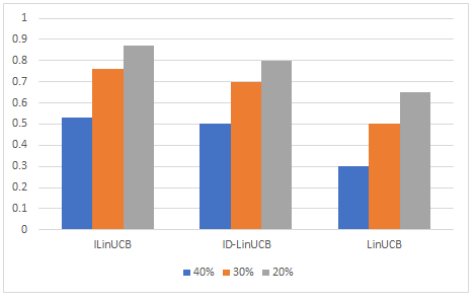


Fig. 2. Total Average Reward for Customer Assistant

In Figure 2 is giving the average rewards we get per iteration. Note that we have run the experiment 1000 times, and we have grouped the iterations that have the same number of sparse rewards. So we get 3 cluster of respectively 20, 30 and 40 percent of sparse reward observed. We can see that the higher the sparsity the lower the rewards for all the algorithm. We can also notice that ILINUCB is having high average rewards compared with the other two algorithms which confirm the usefulness of our approach.

5. CONCLUSION

In real life application of the question answering system, the feedback that is available is most often only a noisy proxy for the actual reward. This paper proposes a new stochastic multi-armed bandit model in which each action has a noisy and a sparse rewards. We have studied this problem in contextual bandit, and propose algorithm that is based on LINUCB framework. We have done a regret analysis of the proposed algorithm and a naive possible solutions to the problem, we showed empirically that our proposed approaches have a better regret than the naive one, we have also verified on several real world data-sets the empirical performance of the proposed algorithm.

6. REFERENCES

[1] T. L. Lai and Herbert Robbins, "Asymptotically efficient adaptive allocation rules," *Advances in Applied Mathematics*, vol. 6, no. 1, pp. 4–22, 1985.

[2] Peter Auer, Nicolò Cesa-Bianchi, and Paul Fischer, "Finite-time analysis of the multiarmed bandit problem," *Machine Learning*, vol. 47, no. 2-3, pp. 235–256, 2002.

[3] Djallel Bouneffouf and Raphaël Féraud, "Multi-armed bandit problem with known trend," *Neurocomputing*, vol. 205, pp. 16–21, 2016.

[4] Robin Allesiardo, Raphaël Feraud, and Djallel Bouneffouf, "A neural networks committee for the contextual bandit problem," in *Neural Information Processing - 21st International Conference, ICONIP 2014, Kuching, Malaysia, November 3-6, 2014. Proceedings, Part I*, 2014, pp. 374–381.

[5] Jonathan P Epperlein, Roman Overko, Sergiy Zhuk, Christopher King, Djallel Bouneffouf, Andrew Cullen, and Robert Shorten, "Reinforcement learning with algorithms from probabilistic structure estimation," *Automatica*, vol. 144, pp. 110483, 2022.

[6] Qinyi Chen, Negin Golrezaei, and Djallel Bouneffouf, "Dynamic bandits with an auto-regressive temporal structure," *arXiv preprint arXiv:2210.16386*, 2022.

[7] Shipra Agrawal and Navin Goyal, "Thompson sampling for contextual bandits with linear payoffs," in *ICML (3)*, 2013, pp. 127–135.

[8] Peter Auer, Nicolò Cesa-Bianchi, Yoav Freund, and Robert E. Schapire, "The nonstochastic multiarmed bandit problem," *SIAM J. Comput.*, vol. 32, no. 1, pp. 48–77, 2002.

[9] Djallel Bouneffouf, Irina Rish, and Charu Aggarwal, "Survey on applications of multi-armed and contextual bandits," in *2020 IEEE Congress on Evolutionary Computation (CEC)*. IEEE, 2020, pp. 1–8.

[10] Djallel Bouneffouf and Charu C Aggarwal, "Survey on applications of neurosymbolic artificial intelligence," *arXiv preprint arXiv:2209.12618*, 2022.

[11] Elliot Nelson, Debarun Bhattacharjya, Tian Gao, Miao Liu, Djallel Bouneffouf, and Pascal Poupart, "Linearizing contextual bandits with latent state dynamics," in *Conference on Uncertainty in Artificial Intelligence*, 2022.

[12] Lihong Li, Wei Chu, John Langford, and Robert E. Schapire, "A contextual-bandit approach to personalized news article recommendation," in *Proceedings of the 19th international conference on World wide web*, USA, 2010, WWW '10, pp. 661–670, ACM.

[13] Wei Chu, Lihong Li, Lev Reyzin, and Robert E. Schapire, "Contextual bandits with linear payoff functions.," in *AISTATS*, Geoffrey J. Gordon, David B. Dunson, and Miroslav Dudik, Eds. 2011, vol. 15 of *JMLR Proceedings*, pp. 208–214, JMLR.

[14] Djallel Bouneffouf, Raphael Feraud, Sohini Upadhyay, Irina Rish, and Yasaman Khazaeni, "Toward optimal solution for the context-attentive bandit problem.," in *IJCAI*, 2021, pp. 3493–3500.

[15] Pratik Gajane, Tanguy Urvoy, and Emilie Kaufmann, "Corrupt bandits," *EWRL*, 2016.

[16] Aditya Grover, Todor Markov, Peter Attia, Norman Jin, Nicolas Perkins, Bryan Cheong, Michael Chen, Zi Yang, Stephen Harris, William Chueh, and Stefano Ermon, "Best arm identification in multi-armed bandits with delayed feedback," in *AISTATS*. 09–11 Apr 2018, vol. 84, pp. 833–842, PMLR.

[17] Baihan Lin, Guillermo A. Cecchi, Djallel Bouneffouf, Jenna M. Reinen, and Irina Rish, "A story of two streams: Reinforcement learning models from human behavior and neuropsychiatry," in *Proceedings of the 19th International Conference on Autonomous Agents and Multiagent Systems, AAMAS '20*, 2020.

[18] Kuan-Hao Huang and Hsuan-Tien Lin, "Linear upper confidence bound algorithm for contextual bandit problem with piled rewards," in *Pacific-Asia Conference on Knowledge Discovery and Data Mining*. Springer, 2016.

[19] Claire Vernade, Olivier Cappé, and Vianney Perchet, "Stochastic bandit models for delayed conversions," *In Conference on Uncertainty in Artificial Intelligence*, 2017.

[20] Claire Vernade, Alexandra Carpentier, Tor Lattimore, Giovanni Zappella, BeyzaErmis, and Michael Brueckner, "Linear bandits with stochastic delayed feedback," 2018.

[21] Zhengyuan Zhou, Renyuan Xu, and Jose Blanchet, "Learning in generalized linear contextual bandits with stochastic delays," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[22] Ramakanth Pasunuru, Han Guo, and Mohit Bansal, "Dorb: Dynamically optimizing multiple rewards with bandits," *arXiv preprint arXiv:2011.07635*, 2020.

[23] Ashkan Sharabiani, Adam Bress, Elnaz Douzali, and Houshang Darabi, "Revisiting warfarin dosing using machine learning techniques," *Comput Math Methods Med*, vol. 2015, pp. 1–9, 07 2015.