Making Recommendations when Users Experience Fatigue

Yunjuan Wang[1] **and Theja Tulabandhula**[2]

[1] Johns Hopkins University, email: ywang509@jhu.edu

[2] University of Illinois at Chicago, email: tt@theja.org

Abstract

In this paper we consider an online recommendation setting, where a platform recommends a sequence of items to its users at every time period. The users respond by selecting one of the items recommended or abandon the platform due to fatigue from seeing less useful items. Assuming a parametric stochastic model of user behavior, which captures positional effects of these items as well as the abandoning behavior of users, the platform's goal is to recommend sequences of items that are competitive to the single best sequence of items in hindsight, without knowing the true user model a priori. Naively applying a stochastic bandit algorithm in this setting leads to an exponential dependence on the number of items. We propose a new Thompson sampling based algorithm with expected regret that is polynomial in the number of items in this combinatorial setting, and performs extremely well in practice.

Introduction

In this paper, we consider the following setting: the platform needs to learn a sequence of items (from a set of N items) by interacting with its users in rounds. In particular, it wants to maximize its expected utility when compared to the best sequence in hindsight. When a user is presented with a sequence of items, they view it from top-to-bottom and at each position, we can have the following stochastic outcomes:

1. The user is satisfied with the current item (perhaps clicks the item's link and navigates to a target page). In this situation, the platform gets a reward.
2. The user is not satisfied with the current item and is willing to look at the next item (for instance, the next notification) if it exists. Note that, it may happen that the user did not select any item and has reached the end of the sequence. In this case, the platform does not get a reward but is also not explicitly penalized.
3. The user has lost interest in the platform (presumably after viewing un-interesting items) and s/he decides to abandons the platform (for instance, by uninstalling the app). In this situation, we ascribe a penalty cost to the platform.

In our setting, the platform can choose both the length of the sequence as well as the order of the items, and this is essentially a combinatorial problem in each round. The recommended sequence of items should balance the penalty of user abandonment versus the upside of user choosing a high revenue item. The probability of a user choosing a high revenue item is not independent of other items in the recommended list. We assume that the aforementioned user behavior has a particular parametric form , whose parameters are not known a priori. Our main contribution is the design of a fatigue-aware online recommendation solution, which we call the *Sequential Bandit Online Recommendation System* (SBORS). SBORS, which is based on Thompson sampling, comes with attractive regret guarantees and makes an ordered list of item recommendations to users by carefully exploring their suitability and exploiting learned information based on previous user feedback.

The key contributions of this paper are as follows: First, we design a Thompson sampling (TS) based algorithm for the online fatigue-aware recommendation problem with unknown user preference and abandonment distributions. Second, we formally present SBORS by modifying the above algorithm with posterior approximation and correlated sampling to control exploration-vs-exploitation tradeoff. We give detailed analysis of SBORS, and prove that the regret upper bound is $C_1N^2\sqrt{NT\log TR}+C_2N\sqrt{T\log TR\cdot\log T}+C_3N/R$ (here C_1, C_2 and C_3 are constants, T is the horizon length, and R is a tunable algorithm parameter that captures exploration-exploitation tradeoff via sampling).

There are many approaches to solve the stochastic bandit problem. One of the mainstream methods is the Upper Confidence Bound (UCB) algorithm (Auer 2002; Bubeck, Cesa-Bianchi, and others 2012; Chen, Wang, and Yuan 2013) (and its many variations). An alternative approach that is different from the UCB family, is the Thompson sampling (TS) approach (Agrawal and Goyal 2012; Russo et al. 2018; Kaufmann, Korda, and Munos 2012). Extensions of these to contextual settings have also been investigated (Li et al. 2010; Cheung and Simchi-Levi 2017) that allow for richer decision making models and algorithms. While some prior work (Wang and Chen 2018; Durand and Gagné 2014) has studied the application of the TS methodology to the stochastic combinatorial multi-armed bandit problem, the combinatorial structure they exploit is not enough to be useful in out setting, or their regret upper bounds or too loose. In our setting, the feasible decisions are sequences of items, which are richer than other objects such as sets.

For a particular combinatorial problem, namely the assortment optimization problem, (Agrawal et al. 2017a) and (Agrawal et al. 2017b) provide UCB and TS based approaches with attractive regret guarantees. Assortment optimization is the task of choosing a set of items that maximizes the expected revenue assuming a user behavior model (similar to our setting). A particular variant of this problem was initially studied in (Rusmevichientong, Shen, and Shmoys 2010; Sauré and Zeevi 2013) and further discussed by (Davis, Gallego, and Topaloglu 2013; Désir, Goyal, and Zhang 2014; Gallego and Topaloglu 2014; Agrawal et al. 2017a; Agrawal et al. 2017b; Agrawal et al. 2016). Since the number of sets is exponential in the number of items, direct application of a MAB solution turns out to be suboptimal. Similar to (Agrawal et al. 2017b), we develop a new algorithm for our online recommendation problem (called SBORS) that comes with attractive regret guarantees. The key difference with assortment optimization is that the problem is polynomially solvable in each round whereas in our case the computational problem in each round is NP-hard. We also consider fatigue, which is not present in assortment optimization. Our analysis builds on the machinery developed by (Agrawal et al. 2017b) and uses correlated sampling to control exploration-exploitation trade-off.

The basic form of sequential choice bandit problem, developed by (Craswell et al. 2008), is a cascade model where a user views search results displayed by web engine from top to bottom and clicks the first attractive one. (Kveton et al. 2015) present an online learning version of the cascade model where the platform receives a reward if a user clicks one item, and solve it using a UCB based algorithm. (Katariya et al. 2016) proposed the DCM bandit, a variant that extends the cascade problem to multiple clicks, and proposed the dcmKL-UCB algorithm. In (Zoghi et al. 2017), the authors present the BatchRank algorithm for a class of click models encompassing the cascading and position-sensitive user behaviors. (Lattimore et al. 2018) build on the work of (Zoghi et al. 2017) and present the TopRank algorithm to find the most attractive list in an online setting. (Cheung, Tan, and Zhong 2019) propose a Thompson Sampling based algorithm to minimize regret under the cascade model. Similarly, the setting in (Cao and Sun 2019) takes the probability of abandoning the platform into consideration, which can be regarded as an extension of the basic cascade model.

Model

Consider a platform containing N different items indexed by i. let its corresponding revenue be r_i if selected. User's intrinsic preference for an item i is denoted by u_i. After viewing each item from a recommended list, the user has a probability p of abandoning the platform, and the occurrence of this event causes the platform to incur a penalty cost c. Note that $r_i, u_i, p, c \in [0, 1]$. We represent the sequence of items at time/round t as $\mathbf{S}^t = (S_1^t, S_2^t, ..., S_m^t)$, where S_i^t denotes the i^{th} item, and m represents the length of the sequence.

After the user at time t sees item i, s/he has three options based on behavior parameters $\mathbf{u}$ and p: (1) The user is satisfied with the item i, then no further items are presented to the user. In this situation, the platform earns revenue r_i. (2) The user is not satisfied with item S_i^t and decides to see the following item S_{i+1}^t in the sequence of items. When the sequence runs out, the user exits the platform. In this situation, the platform will neither earn a reward nor pay a penalty cost. (3) The user is unsatisfied with the platform altogether after looking at some items, and s/he abandons the platform. In this situation, the platform incurs a penalty c.

The behavior parameters $\mathbf{u}$ and p parameterize the following distributions. Consider a random variable W^t following a distribution F_W. W^t measures the t^{th} user's patience, capturing the number of unsatisfied items the user sees without abandoning the platform. In particular, F_W is a geometric distribution with parameter p. Let $q = 1 - p$. Then $q^{k-1}(1 - q)$ denotes the probability that a user abandons the platform after receiving k^{th} unsatisfying item. Further, let $\widetilde{F}_W(k) = P(W > k) = 1 - P(W \leq k) = q^k$ denote the probability that a user does not abandon the platform after receiving the k^{th} unsatisfying item. The probability of each item i being selected is u_i, which is only determined by its content. The probability of each item i being selected when it belongs to the sequence of items $\mathbf{S}$ (dropping the superscript t for simplicity) is denoted as $Pr_i(\mathbf{S})$. $Pr_i(\mathbf{S})$ not only depends on the item's intrinsic value to the user, but also depends on its position and the other items shown before it. The probability of total abandonment is denoted as $Pr_a(\mathbf{S})$, and represents the sum of the probabilities that the platform is abandoned after receiving k unsatisfying items. We denote $U(\mathbf{S}, \mathbf{u}, q)$ as the total utility (payoff) that the platform receives from a given sequence of items $\mathbf{S}$. Define the expected utility as $\mathbb{E}[U(\mathbf{S}; \mathbf{u}, q)] = \sum_{i \in \mathbf{S}} Pr_i(\mathbf{S}) r_i - c Pr_a(\mathbf{S})$. The goal is to find the optimal sequence of items that can optimize $\mathbb{E}[U(\mathbf{S}; \mathbf{u}, q)]$:

$$\begin{aligned} \max_{\mathbf{S}} \quad & \mathbb{E}[U(\mathbf{S}; \mathbf{u}, q)] \\ \text{s.t.} \quad & S_i \cap S_j = \emptyset, \forall i \neq j. \end{aligned} \tag{1}$$

We denote the optimal sequence of items for a given $\mathbf{u}, q$ pair using $\mathbf{S}^* = \arg\max_{\mathbf{S}} \mathbb{E}[U(\mathbf{S}; \mathbf{u}, q)]$, which we assume is unique for simplicity. For a time horizon T, we define the pseudo-regret as below:

$$Reg(T; \mathbf{u}, q) = \mathbb{E}\left[\sum_{t=1}^{T} \mathbb{E}[U(\mathbf{S}^*; \mathbf{u}, q)] - \mathbb{E}[U(\mathbf{S}^t; \mathbf{u}, q)]\right],$$

where $\mathbf{S}^t$ is the sequence offered to the user arriving at time t. Our goal of maximizing expected utility across the rounds is equivalent to minimizing $Reg(T; \mathbf{u}, q)$. Extensions such as making the abandonment probability parameter dependent on the sequence $\mathbf{S}^t$ (such as its length as well as the items in it) are left for future work.

Algorithm

We first describe an algorithm that captures the TS approach. Unfortunately, a direct analysis of this version is difficult, so we modify it suitably to design our proposed algorithm SBORS later on.

Denote $g_i(t)$ as the total number of users selecting item i, and $f_i(t)$ as the total number of users observing item i without selection. Let $T_i(t) = g_i(t) + f_i(t)$. Denote $n_a(t)$ as the number of users who abandon the platform by time t, $n_e(t)$ as the number of times that users do not select an item and do not abandonment by time t. Let $N_q(t) = n_e(t) + n_a(t)$. As shown in (Cao and Sun 2019) (Lemma 5), we can get unbiased estimates of the true parameters as follows:

Lemma 1. *Unbiased estimates:* $\hat{u}_i(t) = \frac{g_i(t)}{T_i(t)}$ *is an unbiased estimator for* u_i *and* $\hat{q}_i(t) = \frac{n_e(t)}{N_q(t)}$ *is an unbiased estimator for* q.

In this version of the algorithm, we maintain a Beta posterior distribution for the selection parameter u_i and the abandonment distribution parameter q, which we update as we observe the user's feedback to our current recommended list. Beta distributions are a natural choice for Bernoulli feedback (likelihood) due to computational gains that can be achieved due to conjugacy. Note that $\mathbf{u}$ and q remain the same across time, which means our recommendation system interact with i.i.d. users in each round. At the initial state, u_i and q are unknown to the platform, r_i and c are known to the platform. For a user arriving at time t, we calculate the current optimal sequence of items based on samples $\mathbf{u}'(t)$ and $q'(t)$. When the sequence of items is shown, the user has three options: (1) select one item and leave the interface; (2) see all the items without selection and abandonment; or (3) abandon the platform. After each round, we update the parameters of the relevant Beta distributions.

Algorithm 1 TS-based algorithm (precursor to SBORS)

```
Initialization: Set $g_i(t) = f_i(t) = 1$ for all $i \in X$;
$n_e(t) = n_a(t) = 1$; and $t = 1$;
while $t \leq T$ do
    (a) Posterior sampling:
    For each item $i = 1, ..., N$, sample $u_i'(t)$ and $q'(t)$
    $u_i'(t) \sim Beta(g_i(t), f_i(t))$
    $q'(t) \sim Beta(n_e(t), n_a(t))$
    (b) Sequence selection:
    Compute $\mathbf{S}^t = \arg\max_{\mathbf{S}} \mathbb{E}[U(\mathbf{S}; \mathbf{u}'(t), q'(t))]$;
    Observe feedback upon seeing the $k_t \leq |\mathbf{S}^t|$ items;
    (c) Posterior update:
    for $j = 1, \cdots, k_t$ do
        Update $(g_{S_j}(t), f_{S_j}(t), n_e(t), n_a(t))$
        $= \begin{cases} (g_{S_j}(t)+1, f_{S_j}(t), n_e(t), n_a(t)) \\ \quad \text{if select and leave} \\ (g_{S_j}(t), f_{S_j}(t)+1, n_e(t)+1, n_a(t)) \\ \quad \text{if not select and not abandon} \\ (g_{S_j}(t), f_{S_j}(t)+1, n_e(t), n_a(t)+1) \\ \quad \text{if not select and abandon} \end{cases}$
    $g_i(t+1) = g_i(t)$, $f_i(t+1) = f_i(t)$ for all $i \in [N]$
    $n_e(t+1) = n_e(t)$, $n_a(t+1) = n_a(t)$
    $t = t + 1$
```

SBORS: Sequential Bandit for Online Recommendation System

Motivated by (Agrawal et al. 2017b), we modify Algorithm 1 by: (a) introducing a posterior approximation by Gaussians, and (b) performing correlated sampling (which boosts variance and allows for a finer exploration-exploitation trade-off). Algorithm 2, Sequential Bandit Online Recommendation System (SBORS), empirically performs similar to Algorithm 1 while being amenable to theoretical analysis.

Posterior approximation: We approximate the posteriors for u_i, q by Gaussian distributions with approximately the same mean and variance as the original Beta distributions. In particular, let

$$\begin{aligned} \hat{u}_i(t) &= \frac{g_i(t)}{g_i(t)+f_i(t)} = \frac{g_i(t)}{T_i(t)}, \\ \hat{\sigma}_{u_i}(t) &= \sqrt{\frac{\alpha \hat{u}_i(t)(1-\hat{u}_i(t))}{T_i(t)+1}} + \sqrt{\frac{\beta}{T_i(t)}}, \end{aligned} \tag{2}$$

$$\begin{aligned} \hat{q}(t) &= \frac{n_e(t)}{n_e(t)+n_a(t)} = \frac{n_e(t)}{N_q(t)}, \text{ and} \\ \hat{\sigma}_q(t) &= \sqrt{\frac{\alpha \hat{q}(t)(1-\hat{q}(t))}{N_q(t)+1}} + \sqrt{\frac{\beta}{N_q(t)}}, \end{aligned} \tag{3}$$

where $\alpha > 0, \beta \geq 2$ are constants, be the means and standard deviations of the approximating Gaussians.

Controlling exploration via correlated sampling: Instead of sampling $\mathbf{u}'$ and q' independently, we correlate them by using a common standard Gaussian sample and transforming it. That is, in the beginning of a round t, we generate a sample from the standard Gaussian $\theta \sim N(0,1)$, and the posterior sample for item i is computed as $\hat{u}_i(t) + \theta\hat{\sigma}_{u_i}(t)$, while the posterior sample for abandonment is computed as $\hat{q}(t) + \theta\hat{\sigma}_q(t)$. This allows us to generate sample parameters for $i = 1, \cdots, N$ that are highly likely to be either simultaneously high or simultaneously low. As a consequence, the parameters corresponding to items in the ground truth $\mathbf{S}^*$, will also be simultaneously high/low. Because correlated sampling decreases the joint variance of the sample, we can counteract by generating multiple Gaussian samples. In particular, we generate R independent samples from the standard Gaussian, $\theta^{(j)} \sim N(0,1)$, $j \in [R]$, and the j^{th} sample of parameters is generated as:

$$u_i'^{(j)} = \hat{u}_i + \theta^{(j)}\hat{\sigma}_{u_i}, \quad \text{ and } q^{'(j)} = \hat{q} + \theta^{(j)}\hat{\sigma}_q.$$

We then use the highest valued samples by simply taking the maximums $u_i'(t) = \max\limits_{j=1,\cdots,R} u_i'^{(j)}(t)$, and $q'(t) = \max\limits_{j=1,\cdots,R} q'^{(j)}(t)$. These are then used in the optimization problem to get $\mathbf{S}^t = \arg\max\limits_{\mathbf{S}} \mathbb{E}[U(\mathbf{S}^t; \mathbf{u}'(t), q'(t))]$.

Algorithm 1 samples from the posterior distribution of $\mathbf{u}$ and q independently in each round, which makes the probability of being optimistic (i.e. the optimal sequence of items $\mathbf{S}^*$ has at least as much reward on the sampled parameters as on the true parameters) exponentially small. We use correlation sampling to ensure that the probability of an optimistic round is high enough.

Regret Analysis for SBORS

Our main result is the following:

Algorithm 2 SBORS algorithm

```
  Initialization: Set $g_i(t) = f_i(t) = 1$ for all $i \in X$;
  $n_e(t) = n_a(t) = 1$; $t = 1$;
  while $t \leq T$ do
    Update $\hat{u}_i(t), \hat{q}(t), \hat{\sigma}_{u_i}(t), \hat{\sigma}_q(t)$ from (2) and (3);
    (a) Correlated sampling:
    for $j = 1, ..., R$ do
      Get $\theta^{(j)} \sim N(0,1)$ and compute $u_i'^{(j)}(t)$,$q'^{(j)}(t)$
    For each $i \leq N$, compute $u_i'(t) = \max_{j=1,\cdots,R} u_i'^{(j)}(t)$ and $q'(t) = \max_{j=1,\cdots,R} q'^{(j)}(t)$.
    (b) Sequence selection: Same as step (b) of Algo. 1.
    (c) Posterior update: Same as step (c) of Algo. 1.
```

Theorem 1. *(**Main Result**) Over T rounds, the regret of SBORS (Algorithm 2) is bounded as:*

$$\begin{aligned} Reg(T; \mathbf{u}, q) \leq{} & C_1 N^2 \sqrt{NT \log TR} \\ & + C_2 N \sqrt{T \log TR \cdot \log T} + \frac{C_3 N}{R}, \end{aligned}$$

where C_1, C_2 and C_3 are constants and R is an algorithm parameter.

Proof Sketch: We provide a proof sketch below and refer the reader to (Wang and Tulabandhula 2019) for a more detailed treatment. The pseudo-regret can be expressed as:

$$Reg(T; \mathbf{u}, q) = \mathbb{E}\left[\sum_{t=1}^{T} \mathbb{E}[U(\mathbf{S}^*; \mathbf{u}, q)] - \mathbb{E}[U(\mathbf{S}^t; \mathbf{u}, q)]\right],$$

where $\mathbf{S}^*$ is the optimal sequence when $\mathbf{u}$ and q are known to the platform, while $\mathbf{S}^t$ is the sequence offered to the user arriving at time t. Adding and subtracting $\sum_{t=1}^{T} \mathbb{E}[U(\mathbf{S}^t, \mathbf{u}'(t), q'(t))]$, we can rewrite the regret as $Reg(T; \mathbf{u}, q) = Reg_1(T, \mathbf{u}, q) + Reg_2(T, \mathbf{u}, q)$ where: $Reg_1(T, \mathbf{u}, q)$ $= \mathbb{E}\left[\sum_{t=1}^{T} \mathbb{E}[U(\mathbf{S}^*; \mathbf{u}, q)] - \mathbb{E}[U(\mathbf{S}^t; \mathbf{u}'(t), q'(t))]\right]$, and $Reg_2(T, \mathbf{u}, q)$ $= \mathbb{E}\left[\sum_{t=1}^{T} \mathbb{E}[U(\mathbf{S}^t; \mathbf{u}'(t), q'(t))] - \mathbb{E}[U(\mathbf{S}^t; \mathbf{u}, q)]\right]$.

We say that a round t is optimistic if the optimal sequence of items $\mathbf{S}^*$ has at least as much reward on the sampled parameters as on the true parameters, i.e. $\mathbb{E}[U(\mathbf{S}^*; \mathbf{u}'(t), q'(t))] \geq \mathbb{E}[U(\mathbf{S}^*; \mathbf{u}, q)]$.

The first term $Reg_1(T, \mathbf{u}, q)$ is the difference between the optimal reward given the true parameters $\mathbf{u}$, q, and the optimal reward of the sampled sequence of items $\mathbf{S}^t$ with respect to the sampled parameters $\mathbf{u}'$, q'. Thus this term would contribute *no regret if the round was optimistic*, as defined above. So, we are left to consider only "non-optimistic" rounds, which we will show they are not too many in number. Thus, we first prove that at least one of our R samples is optimistic with high probability. Then, we also bound the instantaneous regret of any "non-optimistic" round by relating it to the closest optimistic round before it.

The second term $Reg_2(T, \mathbf{u}, q)$ is the difference in the reward of the offer sequence of items $\mathbf{S}^t$ when evaluated on sampled parameters and the true parameters, which can be bounded by the concentration properties of our posterior distributions. The idea is that the expected reward corresponding to the sampled parameters will be close to that on the true parameters. Before elaborating further on the proof details, we first highlight some key lemmas involved in proving Theorem 1 below.

Key Lemmas: To analyze the regret, we first provide the concentration results for the relevant quantities. To be specific, the posterior distributions concentrate around their means, which in turn concentrate around the true parameters.

Lemma 2. *(Concentration bound) For all $i = 1, \cdots, N$, for any $\alpha, \beta, \rho \geq 0$, and $t \in \{1, 2, \cdots, T\}$, we have*

$$P\left(|\hat{u}_i(t) - u_i| \geq \sqrt{\tfrac{\alpha \hat{u}_i(t)(1-\hat{u}_i(t)) \log \rho}{T_i(t)+1}} + \sqrt{\tfrac{\beta \log \rho}{T_i(t)}}\right) \leq \tfrac{2}{\rho^{2\beta}},$$

$$P\left(|\hat{q}(t) - q| \geq \sqrt{\tfrac{\alpha \hat{q}(t)(1-\hat{q}(t)) \log \rho}{N_q(t)+1}} + \sqrt{\tfrac{\beta \log \rho}{N_q(t)}}\right) \leq \tfrac{2}{\rho^{2\beta}}.$$

Lemma 3. *For any $t \leq T$ and $i \in \{1, \cdots, N\}$, we have for any $r > 1$,*

$$P(|u_i'(t) - \hat{u}_i(t)| > 4\hat{\sigma}_{u_i}(t)\sqrt{\log rR}) \leq \tfrac{1}{r^8 R^7} \ , \textit{ and}$$

$$P(|q'(t) - \hat{q}(t)| > 4\hat{\sigma}_q(t)\sqrt{\log rR}) \leq \tfrac{1}{r^8 R^7} \ ,$$

where $\hat{\sigma}_{u_i}(t)$, $\hat{\sigma}_q(t)$, R, $u_i'(t)$, $q'(t)$, $\hat{u}$, $\hat{q}$ are defined in Section .

Next we establish two important properties of the optimal expected payoff. The first property is referred to as restricted monotonicity. Simply put, with the optimal sequence of items $\mathbf{S}_v^*$ determined under some parameters $\mathbf{v}$ and q_v, its expected payoff is no larger than the payoff under the same sequence of items $\mathbf{S}_v^*$ when preference parameter $\mathbf{w}$ and the abandonment parameter q_w are element-wise larger than $\mathbf{v}$ and q_v. The second property is a *Lipschitz* style bound on the deviation of the expected payoff with change in the parameters $\mathbf{v}$ and q_v. To be specific, the difference between the two expected payoffs is bounded by a linear sum of the items' preference and abandonment parameters.

Lemma 4. *Suppose $\mathbf{S}_v^*$ is an optimal sequence of items given $\mathbf{v}$ and q_v. That is, $\mathbf{S}_v^* \in \arg\max \mathbb{E}[U(\mathbf{S}, \mathbf{v}, q_v)]$.*

Then for any $\mathbf{v}, \mathbf{w} \in [0,1]^N$, $q_v, q_w \in [0,1]$, we have

1. (Restricted Monotonicity) If $v_i \leq w_i$ for all $i \in [N]$, and $q_v \leq q_w$, then $\mathbb{E}[U(\mathbf{S}_v^*; \mathbf{w}, q_w)] \geq \mathbb{E}[U(\mathbf{S}_v^*; \mathbf{v}, q_v)]$.

2. (Lipschitz)

$$|\mathbb{E}[U(\mathbf{S}_v^*, \mathbf{v}, q_v)] - \mathbb{E}[U(\mathbf{S}_v^*, \mathbf{w}, q_w)]| \leq \sum_{i \in \mathbf{S}_v^*} (2|v_i - w_i| + (N+1)|q_v - q_w|) \, .$$

From Lemma 2, 3 and 4, we can prove that the difference between the expected payoff of the offered sequence $\mathbf{S}^t$ corresponding to the sampled parameters and the true parameters becomes smaller as time increases.

Lemma 5. *For any round $t \leq T$, we have*

$$\mathbb{E}\left\{\mathbb{E}[U(\mathbf{S}^t, \mathbf{u}'(t), q'(t))] - \mathbb{E}[U(\mathbf{S}^t, \mathbf{u}, q)]\right\} \leq \mathbb{E}\left[C_1' \sum_{i \in \mathbf{S}^t} \sqrt{\frac{\log TR}{T_i(t)}} + C_2'(N+1)\sqrt{\frac{\log TR}{N_q(t)}}\right],$$

where C_1' and C_2' are universal constants.

We will now discuss how these lemmas can be put together to bound $Reg_1(T, \mathbf{u}, q)$ and $Reg_2(T, \mathbf{u}, q)$. **Bounding the first term** $Reg_1(T, \mathbf{u}, q)$: Since $\mathbf{S}^t$ is an optimal sequence of items for the sampled parameters, we have $\mathbb{E}[U(\mathbf{S}^t; \mathbf{u}'(t), q'(t))] \geq \mathbb{E}[U(\mathbf{S}^*; \mathbf{u}, q)]$ if round t is optimistic. This suggests that as the number of optimistic round increases, the term $Reg_1(T, \mathbf{u}, q)$ decreases.

Next, we prove that there are only a limited number of non-optimistic rounds (*this is a key step*). Using a tail bound for the Gaussian distribution, we can control the probability mass associated with the event that a sampled parameter $u_i'^{(j)}(t)$ for any item i will exceed the posterior mean by a few standard deviations. Since our Gaussian posterior's mean is equal to the unbiased estimate $\hat{u}_i$, and its standard deviation is close to the expected deviation of estimate $\hat{u}_i$ from the true parameter u_i, we can conclude that any sampled parameter $u_i'^{(j)}(t)$ will be optimistic with at least a constant probability, i.e., $u_i'^{(j)}(t) \geq u_i$. The same reasoning also holds for $q'^{(j)}(t)$. However, for an optimistic round, sampled parameters for all items in $\mathbf{S}^*$ needs to be optimistic. This is where the correlated sampling aspect of SBORS is crucially utilized. Using the dependence structure between samples for items in $\mathbf{S}^*$, and the variance boosting provided by the sampling of R independent copies, we prove an upper bound of roughly $O(1/R)$ on the number of consecutive rounds between two optimistic rounds. Lemma 6 formalizes this intuition.

Lemma 6. *(Spacing of optimistic rounds) For any $p \in [1, 2]$, we have*

$$\mathbb{E}^{1/p}\left[|\varepsilon^{An}(\tau)|^p\right] \leq \frac{e^{12}}{R} + (C_3'N)^{1/p} + C_4'^{1/p}$$

where C_3' and C_4' are constants. $\varepsilon^{An}(\tau)$ is defined as the group of rounds after an optimistic round τ

and before the next consecutive optimistic round. A formal definition of optimistic round is in Section .

Next, we bound the individual contribution of any "non-optimistic" round t by relating it to the closest optimistic round τ before it. By the definition of an optimistic round,

$$\begin{aligned}&\mathbb{E}[U(\mathbf{S}^*, \mathbf{u}, q)] - \mathbb{E}[U(\mathbf{S}^t, \mathbf{u}'(t), q'(t))]\\ &\le \mathbb{E}[U(\mathbf{S}^\tau, \mathbf{u}(\tau), q(\tau))] - \mathbb{E}[U(\mathbf{S}^t, \mathbf{u}'(t), q'(t))],\end{aligned}$$

and by the choice of $\mathbf{S}_t$ we get:

$$\begin{aligned}&\mathbb{E}[U(\mathbf{S}^\tau, \mathbf{u}(\tau), q(\tau))] - \mathbb{E}[U(\mathbf{S}^t, \mathbf{u}'(t), q'(t))]\\ &\le \mathbb{E}[U(\mathbf{S}^\tau, \mathbf{u}(\tau), q(\tau))] - \mathbb{E}[U(\mathbf{S}^\tau, \mathbf{u}'(t), q'(t))].\end{aligned}$$

What remains to be shown is a bound on the difference in the expected payoff of $\mathbf{S}^\tau$ for $\mathbf{u}(\tau), q(\tau)$ and for $\mathbf{u}'(t), q'(t)$. Over time, as the posterior distributions concentrate around their means, which in turn concentrate around the true parameters, we can show that this difference becomes smaller. As a result, Reg_1 can be bounded as: $Reg_1(T, \mathbf{u}, q) \le O(N\sqrt{T}\log TR \log T) + O(N/R)$.

Bounding the second term $Reg_2(T, \mathbf{u}, q)$: Similar to the discussion above, using the Lipschitz property (Lemma 4) and Lemma 5, this term can be bounded as: $Reg_2(T, \mathbf{u}, q) \le O(N^2\sqrt{NT \log TR})$. Overall, the above analysis on Reg_1 and Reg_2 implies the following bound on the overall regret:

$$\begin{aligned}&Reg(T; \mathbf{u}, q)\\ &\quad \le C_1 N^2 \sqrt{NT \log TR} + C_2 N\sqrt{T \log TR \cdot \log T}.\end{aligned}$$

Comparison with UCB-V algorithm

In this section we compare SBORS with UCB-V (Audibert, Munos, and Szepesvári 2009) due to the similarities in the way both these techniques maintain estimated means and variances ($\hat{u}_i(t), \hat{q}(t), \hat{\sigma}_{u_i}(t)$ and $\hat{\sigma}_q(t)$). The UCB-V algorithm, designed for the vanilla MAB setting, takes the variance of the different arms into consideration while choosing the next action. By estimating the variance explicitly, UCB-V has the ability to reduce the exploration (bonus) budget spent on certain arms, drastically reducing the regret incurred. In particular, it can be shown that the regret of UCB-V is smaller if the variance of suboptimal items is small.

Although UCB-V algorithm shares some similarities with SBORS algorithm since both these consider variance of the parameters involved, they are fundamentally different. In the SBORS algorithm, parameters $\mathbf{u}, q$ are random variables that are sampled from Gaussian distributions, whereas for the UCB-V algorithm, these are fixed unknowns and their estimates are maintained as $\hat{u}_i, \hat{q}$. SBORS achieves exploration via sampling, whereas UCB-V achives exploration via explicit bonus terms and does not rely on randomization.

Nonetheless, we design an extension of UCB-V that uses variance estimates to improve recommendations in our setting based on ideas from (Cao and Sun 2019) and (Audibert, Munos, and Szepesvári 2009). This algorithm (Algorithm 3) is different from the algorithm proposed by (Cao and Sun 2019) in that it considers the variance of the parameters related to different items, as shown in Equation (4). The update for q (5) is left unchanged:

$$u_{i,t}^{UCB} = \hat{u}_i(t) + \sqrt{\frac{2\mathrm{Var}(\hat{u}_i(t)) \log t}{T_i(t)}} + \frac{b \log t}{T_i(t)}, \tag{4}$$

and

$$q_t^{UCB} = \hat{q}(t) + \sqrt{\frac{2 \log t}{N_q(t)}}, \tag{5}$$

where $\hat{u}_i(t)$, $\hat{q}(t)$ can be computed by Lemma 1, $\mathrm{Var}(\hat{u}_i(t))$ is the estimated variance of $\hat{u}_i(t)$ at time t, and b is the upper bound on the support of u_is.

Algorithm 3 UCB-V algorithm

```
Initialization: Set $u_{i,0}^{UCB} = 1$ for all $i \in [N]$ and $q_0^{UCB} = 1$. Set $c_i(t) = f_i(t) = 1$ for all $i \in [N]$, $n_e(t) = n_a(t) = 1$; and $t = 1$.
while $t \le T$ do
    Compute $\mathbf{S}^t = \arg\max_{\mathbf{S}} \mathbb{E}[U(\mathbf{S}; \mathbf{u}_{t-1}^{UCB}, q_{t-1}^{UCB})]$
    Offer sequence $\mathbf{S}^t$, observe feedback of user who sees $k_t \le |\mathbf{S}^t|$ items.
    for $i = 1, \cdots, [N]$ do
        Update $u_{I(i),t}^{UCB}$ according to Equation (4).
    Update $c_i(t)$, $f_i(t)$, $ne(t)$ and $na(t)$.
    Update $q^{UCB}$ according to Equation (5).
    $t = t + 1$.
```

Conclusion

In this paper, we present a new Thompson sampling based algorithm for making recommendations where users experience fatigue. We use techniques such as posterior approximation using Gaussians, correlate sampling and variance boosting to control the exploration-exploitation trade-off and derive rigorous regret upper bounds. Our bounds depend polynomially on the number of items and sublinearly on the time horizon ($C_1 N^2 \sqrt{NT \log TR} + C_2 N \sqrt{T \log TR \cdot \log T} + C_3 N/R$). Future directions include extensive experiments, generalizing the abandonment model, tackling the computational complexity of the combinatorial problem in each round, tightening the regret upper bound, and extending the machinery to recommendation systems with a variety of other user behavior models.

References

[Agrawal and Goyal 2012] Agrawal, S., and Goyal, N. 2012. Analysis of Thompson sampling for the multi-armed bandit problem. In *Conference on Learning Theory*, 39–1.

[Agrawal et al. 2016] Agrawal, S.; Avadhanula, V.; Goyal, V.; and Zeevi, A. 2016. A near-optimal exploration-exploitation approach for assortment selection. In *Proceedings of the 2016 ACM Conference on Economics and Computation*, 599–600. ACM.

[Agrawal et al. 2017a] Agrawal, S.; Avadhanula, V.; Goyal, V.; and Zeevi, A. 2017a. Mnl-bandit: A dynamic learning approach to assortment selection. *CoRR* abs/1706.03880.

[Agrawal et al. 2017b] Agrawal, S.; Avadhanula, V.; Goyal, V.; and Zeevi, A. 2017b. Thompson sampling for the mnl-bandit. In Kale, S., and Shamir, O., eds., *Proceedings of the 30th Conference on Learning Theory, COLT 2017, Amsterdam, The Netherlands, 7-10 July 2017*, volume 65 of *Proceedings of Machine Learning Research*, 76–78. PMLR.

[Audibert, Munos, and Szepesvári 2009] Audibert, J.-Y.; Munos, R.; and Szepesvári, C. 2009. Exploration–exploitation tradeoff using variance estimates in multi-armed bandits. *Theoretical Computer Science* 410(19):1876–1902.

[Auer 2002] Auer, P. 2002. Using confidence bounds for exploitation-exploration trade-offs. *Journal of Machine Learning Research* 3(Nov):397–422.

[Bubeck, Cesa-Bianchi, and others 2012] Bubeck, S.; Cesa-Bianchi, N.; et al. 2012. Regret analysis of stochastic and nonstochastic multi-armed bandit problems. *Foundations and Trends® in Machine Learning* 5(1):1–122.

[Cao and Sun 2019] Cao, J., and Sun, W. 2019. Dynamic learning of sequential choice bandit problem under marketing fatigue. *The Thirty-Third AAAI Conference on Artificial Intelligence (AAAI-19)*.

[Chen, Wang, and Yuan 2013] Chen, W.; Wang, Y.; and Yuan, Y. 2013. Combinatorial multi-armed bandit: General framework and applications. In *International Conference on Machine Learning*, 151–159.

[Cheung and Simchi-Levi 2017] Cheung, W. C., and Simchi-Levi, D. 2017. Thompson sampling for online personalized assortment optimization problems with multinomial logit choice models.

[Cheung, Tan, and Zhong 2019] Cheung, W. C.; Tan, V.; and Zhong, Z. 2019. A thompson sampling algorithm for cascading bandits. In Chaudhuri, K., and Sugiyama, M., eds., *The 22nd International Conference on Artificial Intelligence and Statistics, AISTATS 2019, 16-18 April 2019, Naha, Okinawa, Japan*, volume 89 of *Proceedings of Machine Learning Research*, 438–447. PMLR.

[Craswell et al. 2008] Craswell, N.; Zoeter, O.; Taylor, M.; and Ramsey, B. 2008. An experimental comparison of click position-bias models. In *Proceedings of the 2008 international conference on web search and data mining*, 87–94. ACM.

[Davis, Gallego, and Topaloglu 2013] Davis, J.; Gallego, G.; and Topaloglu, H. 2013. Assortment planning under the multinomial logit model with totally unimodular constraint structures. *Work in Progress*.

[Désir, Goyal, and Zhang 2014] Désir, A.; Goyal, V.; and Zhang, J. 2014. Near-optimal algorithms for capacity constrained assortment optimization.

[Durand and Gagné 2014] Durand, A., and Gagné, C. 2014. Thompson sampling for combinatorial bandits and its application to online feature selection. In *Workshops at the Twenty-Eighth AAAI Conference on Artificial Intelligence*.

[Gallego and Topaloglu 2014] Gallego, G., and Topaloglu, H. 2014. Constrained assortment optimization for the nested logit model. *Management Science* 60(10):2583–2601.

[Katariya et al. 2016] Katariya, S.; Kveton, B.; Szepesvari, C.; and Wen, Z. 2016. Dcm bandits: Learning to rank with multiple clicks. In *International Conference on Machine Learning*, 1215–1224.

[Kaufmann, Korda, and Munos 2012] Kaufmann, E.; Korda, N.; and Munos, R. 2012. Thompson sampling: An asymptotically optimal finite-time analysis. In *International Conference on Algorithmic Learning Theory*, 199–213. Springer.

[Kveton et al. 2015] Kveton, B.; Szepesvari, C.; Wen, Z.; and Ashkan, A. 2015. Cascading bandits: Learning to rank in the cascade model. *arXiv preprint arXiv:1502.02763*.

[Lattimore et al. 2018] Lattimore, T.; Kveton, B.; Li, S.; and Szepesvari, C. 2018. Toprank: A practical algorithm for online stochastic ranking. In *Advances in Neural Information Processing Systems*, 3945–3954.

[Li et al. 2010] Li, L.; Chu, W.; Langford, J.; and Schapire, R. E. 2010. A contextual-bandit approach to personalized news article recommendation. In *Proceedings of the 19th international conference on World wide web*, 661–670. ACM.

[Rusmevichientong, Shen, and Shmoys 2010] Rusmevichientong, P.; Shen, Z.-J. M.; and Shmoys, D. B. 2010. Dynamic assortment optimization with a multinomial logit choice model and capacity constraint. *Operations research* 58(6):1666–1680.

[Russo et al. 2018] Russo, D. J.; Van Roy, B.; Kazerouni, A.; Osband, I.; Wen, Z.; et al. 2018. A tutorial on Thompson sampling. *Foundations and Trends® in Machine Learning* 11(1):1–96.

[Sauré and Zeevi 2013] Sauré, D., and Zeevi, A. 2013. Optimal dynamic assortment planning with demand learning. *Manufacturing & Service Operations Management* 15(3):387–404.

[Wang and Chen 2018] Wang, S., and Chen, W. 2018. Thompson sampling for combinatorial semi-bandits. *arXiv preprint arXiv:1803.04623*.

[Wang and Tulabandhula 2019] Wang, Y., and Tulabandhula, T. 2019. Making recommendations when users experience fatigue. *Arxiv preprint abs/1901.07734*.

[Zoghi et al. 2017] Zoghi, M.; Tunys, T.; Ghavamzadeh, M.; Kveton, B.; Szepesvari, C.; and Wen, Z. 2017. Online learning to rank in stochastic click models. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, 4199–4208. JMLR. org.