Improving Mobile Maternal and Child Health Care Programs: Collaborative Bandits for Time slot selection

Soumyabrata Pal*
Adobe
soumyabratapal13@gmail.com

Milind Tambe
Google Research
milindtambe@google.com

Arun Suggala
Google Research India
arunss@google.com

Karthikeyan Shanmugam
Google Research India
karthikeyanvs@google.com

Aparna Taneja
Google Research India
aparnataneja@google.com

ABSTRACT

Maternal and child health is a global priority, reflected in the UN Sustainable Development Goal 3.1. Mobile health (mHealth) programs, using automated voice messages, are a vital tool for NGOs to disseminate health information in underserved communities. However, these programs face challenges: limited beneficiary phone access and unknown time preferences hinder timely outreach, leading to poor engagement. We address this by formulating the time preference inference problem as a multi-agent multi-armed bandit optimization problem, where beneficiaries are modeled as agents, and time slots as arms. We introduce a novel online collaborative filtering framework that infers preferred time slots by collaborating across beneficiaries to quickly identify their preferred time slots.

To highlight the scope and impact of this problem, we are working with Kilkari, the world's largest maternal and child mHealth program serving millions in India every week. Kilkari faces substantial reattempt costs to improve call answer rates. Through extensive experiments on real-world data obtained from Kilkari, we demonstrate that our collaborative bandit framework significantly outperforms both existing policies used by the NGO, and popular non-collaborative bandit algorithms (e.g., Upper Confidence Bound), both in terms of number of call retries, saving critical bandwidth that enables wider outreach, and by rapidly learning optimal time slots, improving beneficiary engagement and retention.

KEYWORDS

AI for Social Good, Maternal and Child Healthcare, Bandit Optimization, Collaboration, Matrix Completion

ACM Reference Format:
Soumyabrata Pal, Milind Tambe, Arun Suggala, Karthikeyan Shanmugam, and Aparna Taneja. 2024. Improving Mobile Maternal and Child Health Care Programs: Collaborative Bandits for Time slot selection. In *Proc. of the 23rd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2024), Auckland, New Zealand, May 6 – 10, 2024*, IFAAMAS, 17 pages.

*Work done as a visiting researcher at Google Research India

Proc. of the 23rd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2024), N. Alechina, V. Dignum, M. Dastani, J.S. Sichman (eds.), May 6 – 10, 2024, Auckland, New Zealand. © 2024 International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org).

1 INTRODUCTION

Maternal mortality is unacceptably high in several parts of the world. In 2020, an estimated 287,000 women died from preventable causes related to pregnancy and childbirth [30]. Consequently, the World Health Organization (WHO) has made improving maternal health one of its top priorities. As part of WHO's Sustainable Development Goals (SDGs), countries have committed to reducing the global maternal mortality ratio to less than 70 per 100,000 live births by 2030 and end preventable deaths of newborns and children under 5 years of age [31]. In line with this goal, several non-governmental organizations (NGOs) are leveraging mobile health programs extensively to disseminate critical health information [3, 17, 26, 28] economically due to the widespread availability of cellphones.

Kilkari is the largest maternal and child mHealth program in the world [3] which is implemented by the NGO ARMMAN in partnership with the Ministry of Health and Family Welfare of India, and currently has 3.2 million active users. Kilkari uses free pre-recorded voice calls to deliver vital preventive care information on maternal and infant health to pregnant women and new mothers. To ensure that beneficiaries listen to these messages in a timely manner, it is vital to reach out to them at the right time. The reality is, practical problems such as limited access to phones due to shared family phone for many women, working hours, house chore responsibilities significantly affect the likelihood of engagement at a given time slot [6]. And hence, sending these automated voice calls at an inconvenient/wrong time leads to poor listenership.

In fact, consistent low listenership of the calls can even lead to beneficiaries being dropped from the program. To address this, the NGO re-attempts sending voice messages multiple times in a week until the call is answered. Despite this listenership remains low with almost 50% of the economically weakest beneficiaries requiring more than 6 attempts on average [27] and on average 23% being unreachable despite multiple attempts [23]. [27] pointed out the positive impact of listening to Kilkari messages on health outcomes, particularly among the most marginalised who have the most to benefit from this program, and have the least access to resources. However, despite the known advantages of sustained high listenership, the scale of Kilkari's operations makes it difficult to gather individual time preferences, or demographic information that could help predict those preferences [34]. This makes the problem of identifying good time-slots even more challenging and critical to the efficacy of the program.

Randomized policies such as the one's currently deployed by Kilkari pick a time slot at random, and call the beneficiary. However, such policies tend to be sub-optimal as they do not adapt to beneficiary's preferences (see Section 4 for empirical evidence), and are a suboptimal use of the limited calling bandwidth [6]. To help NGOs overcome this challenge, we study the problem of time slot selection from the perspective of bandit optimization. To be precise, we formulate the problem as a multi-agent multi-armed bandit optimization problem where the beneficiaries are agents and time slots are arms. Every day, one arm of every agent gets pulled and we observe the feedback/reward corresponding to the arm. Our goal is to quickly identify good time slots of each beneficiary, which can be quantified using the following two metrics: (a) average number of retries before a successful pick-up, (b) time to learn these preferences. Numerous algorithms such as Upper Confidence Bound (UCB) [5], Thompson Sampling (TS) [1, 36], Successive Elimination [4] have been proposed to solve the problem in the single agent setting. While these algorithms are optimal for a single agent, they tend to be sub-optimal in the multi-agent setting where one can collaborate across similar agents to identify good arms much faster, which is critical to retain and engage beneficiaries with the program.

In this work, we consider collaborative bandit algorithms for solving the time slot selection problem. Our algorithms try to simultaneously identify similar agents and collaborate across them to learn their arm preferences. We would like to highlight that we do this without access to any features of the agent. This is achieved via a reduction of the collaborative bandit problem to low-rank matrix completion problem where one tries to reconstruct a matrix from a small subset of its entries [18, 22]. In this work, we propose two novel algorithms: Greedy Matrix Completion (MC), and Phased MC [19, 32]. In Greedy MC, we first collect exploratory data, then perform matrix completion with the collected data to estimate the arm preferences, and pull the best estimated arm for each agent in the rest of the rounds. One of the key novelties in Greedy MC, compared to existing works [19, 32], is that we bring in variance reduction techniques [8] into our algorithm which makes it robust to noise and improves its performance in practice. While Greedy MC is effective in practice, the random exploration it performs in early stages can hurt the user experience, which may lead to dropouts from the program. To address this, we introduce Phased MC, a novel bandit algorithm that operates in phases. Instead of having a lengthy random exploration phase at the beginning, it combines exploration and exploitation throughout the program duration. Within each phase, we use Boltzmann exploration based on the estimated reward matrix from previous phases [12]. At the end of every phase, we use robust matrix completion to revise the estimated reward matrix. In Section 2, we provide a more detailed comparison between our algorithms and existing collaborative bandit algorithms.

In Section 4, we evaluate our algorithms on an anonymized real-world dataset obtained from Kilkari, collected over a period of one year for $200k$ beneficiaries. The Kilkari program has 7 time slots in a day during which the calls can be placed. Moreover, calls are placed in a week at most 9 times until there is a pickup by the beneficiary. Considering each slot (irrespective of the day) as an arm, in our experiments, we demonstrate that the MC based bandit algorithms (Greedy MC and Phased MC) achieve a reduction of at least 27% in regret over non collaborative policies like UCB. Furthermore, our MC based algorithms obtain **>30%** reduction in average number of retries per week over random policy (that is in use by Kilkari) and **>8%** reduction over the non collaborative UCB policy, for **42%** of the beneficiaries. When we account for the weekend/weekday effect and increase our arm space to 14 (7 for weekends and 7 for weekdays), we get **>45%** reduction in the number of retries over random policy and **>25%** reduction over UCB policy for approximately **76%** of the users. *An optimistic estimate for the* 14 *time slots case shows that if one were to equalize the total number of calls to what the UCB policy would make, we could potentially onboard **56%** more beneficiaries (based on the plot in Fig. 2c).*

2 RELATED WORK

Maternal Healthcare. Restless Multi-Armed Bandits have been used for improving maternal healthcare by providing solutions for limited resource allocation particularly where NGO's serving under-served communities are operating with limited resources [26, 29, 37]. In contrast, this paper focuses on stochastic MABs for time slot planning.

Bandits. Multi-armed bandits (MAB) and other bandit optimization problems have been widely studied in recent decades. UCB [5], TS [1, 36], Phased Elimination [24, 35] are some of the most popular algorithms for regret minimization in MABs. Recent works have also studied best-arm identification in MABs and provided (near) optimal algorithms [2, 15, 20, 21].

Collaborative Bandits. Collaborative bandit optimization has recently gained significant attention due to its applicability in modern recommendation systems where millions of users interact with the system daily [9, 14, 16, 19, 32]. However, optimal algorithms for this problem are only known under certain special settings. [32] derived a regret optimal algorithm under the assumption that users can be grouped into a small number of latent clusters. [19] assumed the agents $\times$ arms reward matrix has rank 1 and derived an algorithm which achieves optimal regret. However, the assumptions made in both these works rarely hold in practice. [14] developed a heuristic, alternating linear bandits algorithm for low-rank reward matrices. However, this algorithm has poor performance in practice as the reward matrices in real-world are only *approximately* low-rank. One of key technical contributions of our work is to develop algorithms for collaborative bandits under *approximate* low-rank assumptions. Our Greedy MC algorithm is inspired by the greedy algorithm of [19]. The main novelty in Greedy MC comes from the variance reduction technique we introduce to make it robust to noise. Our Phased MC algorithm is inspired by the phased elimination algorithms of [19, 32]. But unlike the algorithms of [19, 32] which assume rank 1 reward matrix or cluster structure among agents, we work with a more general approximate low-rank assumption.

3 PROBLEM FORMULATION AND ALGORITHMS

Notation. We write $[m]$ to denote the set $\{1, 2, \ldots, m\}$. For a vector $\mathbf{v} \in \mathbb{R}^m$, $\mathbf{v}_i$ denotes the i^{th} element; for any set $\mathcal{U} \subseteq [m]$, let $\mathbf{v}_{\mathcal{U}}$ denote the vector $\mathbf{v}$ restricted to the indices in $\mathcal{U}$. Similarly, for

$\mathbf{A} \in \mathbb{R}^{m\times n}$, $\mathbf{A}_{ij}, \mathbf{A}_i$ denotes the (i,j)-th element and the i^{th} row of $\mathbf{A}$ respectively. For any set $\mathcal{U} \subseteq [m], \mathcal{V} \subseteq [n]$, $\mathbf{A}_{\mathcal{U},\mathcal{V}}$ denotes $\mathbf{A}$ restricted to the rows in $\mathcal{U}$ and columns in $\mathcal{V}$. Let $\|\mathbf{A}\|_\infty$ denote absolute value of the largest entry in matrix $\mathbf{A}$. $\text{Ber}(p)$ denotes the binary random variable that is 1 with probability p (0 with probability $1-p$). $\mathbb{E}X$ denotes expectation of random variable X.

Problem Setting: We have M beneficiaries and N slots (distinct intervals of time during the day) where calls can be placed to each beneficiary. Furthermore, there are T rounds - each round corresponding to a day - at which the service provider can make calls to as many beneficiaries as possible. In Kilkari, we have $M \approx 10^6$, $N = 7$, and $T \approx 300$. Our goal is to design a sequential decision making algorithm $\mathcal{A}$ that recommends the service provider appropriate slots to call the beneficiaries. To do this, $\mathcal{A}$ relies on the data obtained via calls made until the previous day and aims to quickly find the preferred time slot for every beneficiary. We model the *unknown* preferences of the beneficiaries towards the distinct slots using the following two matrices:

(1) **Pick-up matrix.** We let $\mathbf{P} \in \mathbb{R}^{\mathsf{M}\times\mathsf{N}}$ denote the pick-up probabilities of beneficiaries. Here $\mathbf{P}_{ij}$ denotes the probability of beneficiary i picking-up a call at slot j.

(2) **Engagement matrix.** We let $\mathbf{E} \in \mathbb{R}^{\mathsf{M}\times\mathsf{N}}$ denote the engagement probabilities of beneficiaries, where $\mathbf{E}_{ij}$ denotes the probability of beneficiary i picking-up and engaging with the call for at least 25% of the message duration at slot j.

In this work we assume that $\mathbf{P}, \mathbf{E}$ are approximately low-rank matrices. We note that this assumption typically holds in many domains; *e.g.,* movie preferences [7], genomics [10]. This assumption also holds in Kilkari, where we observed that $\mathbf{P}, \mathbf{E}$ are approximately rank 2 matrices (see Figure 11).

We model the problem of learning optimal time slots for beneficiaries as a multi-agent, multi-armed bandit optimization problem with M agents (beneficiaries), N arms (time-slots) and T rounds (days). For simplicity, we assume that at each round, the service provider has sufficient capacity to call all the agents. In practice, there can exist limitations on the number of beneficiaries that can be called at a certain time slot or on a certain day. As we show later in the paper, some of these constraints can be easily incorporated into our algorithms (Section 4.2).

Suppose, for beneficiary u at round t, algorithm $\mathcal{A}$ recommends the slot $\rho_u(t)$ for making the call and observes a noisy feedback - pick-up and engagement. The beneficiary u will pick-up the call with probability $\mathbf{P}_{u\rho_u(t)}$ and conditioned on the call being picked up, the beneficiary will engage with probability $\mathbf{E}_{u\rho_u(t)} \times [\mathbf{P}_{u\rho_u(t)}]^{-1}$. We will also define the functions $\pi_p : [\mathsf{M}] \to [\mathsf{N}]$ and $\pi_e : [\mathsf{M}] \to [\mathsf{N}]$ that takes as input a beneficiary and maps it to the slot with the highest pick-up probability and engagement probability respectively for the chosen beneficiary. Next, we define two notions of regret, namely $\text{Reg}_{\text{pick-up}}(\mathsf{T})$ and $\text{Reg}_{\text{engage}}(\mathsf{T})$:

$$\text{Reg}_{\text{pick-up}}(\mathsf{T}) = \sum_{u\in[\mathsf{M}]} \Big(\mathsf{T}\mathbf{P}_{u\pi_p(u)} - \mathbb{E}_{\mathcal{A}} \sum_{t\in[\mathsf{T}]} \mathbf{P}_{u\rho_u(t)}\Big) \quad (1)$$

$$\text{Reg}_{\text{engage}}(\mathsf{T}) = \sum_{u\in[\mathsf{M}]} \Big(\mathsf{T}\mathbf{E}_{u\pi_e(u)} - \mathbb{E}_{\mathcal{A}} \sum_{t\in[\mathsf{T}]} \mathbf{R}_{u\rho_u(t)}\Big). \quad (2)$$

Intuitively, $\text{Reg}_{\text{pick-up}}(\mathsf{T})/\mathsf{TM}$ measures how much the average call pick-up rate of an algorithm differs from the call pick-up rate of the best possible policy. Note that, even though we want to improve engagement more than call pick-up, it is much harder to do so because very few people engage with the calls, and consequently, the data on engagement is very limited. Nonetheless, in Section 4.1.2 we show that one could improve engagement rate by smartly combining the pick-up and engagement data.

3.1 Collaborative Algorithms

Our collaborative bandit algorithms rely on an offline low rank matrix completion oracle $\mathcal{O}$. For an unknown matrix $\mathbf{Z} \in \mathbb{R}^{m\times n}$, $\mathcal{O}$ takes a subset of noisy observations of the matrix at positions Ω ($\{\mathbf{M}_{ij}\}_{(i,j)\in\Omega}$) as input, and returns an estimate $\widehat{\mathbf{Z}}$ of $\mathbf{Z}$. To implement this oracle, we minimize the following nuclear norm regularized objective [13]:

$$\text{minimize}_{\widehat{\mathbf{Z}}} \sum_{(i,j)\in\Omega} (\mathbf{M}_{ij} - \widehat{\mathbf{Z}}_{ij})^2 + \lambda \left\|\widehat{\mathbf{Z}}\right\|_\star . \quad (3)$$

Here $\lambda > 0$ is the regularization parameter and $\left\|\widehat{\mathbf{Z}}\right\|_\star$ denotes the nuclear norm of $\widehat{\mathbf{Z}}$. We note that this is a very popular technique for matrix completion, and comes with strong theoretical guarantees [13].

3.1.1 Greedy Matrix Completion (MC). In this section, we present our first algorithm, Greedy MC (Algorithm 1). The key idea here is to partition the T rounds into two phases - *exploration* phase spanning the first $\mathsf{T}_{\text{explore}}$ rounds and *exploitation* phase spanning the next $\mathsf{T} - \mathsf{T}_{\text{explore}}$ rounds. In the exploration phase, we randomly select a slot for each beneficiary and place a call in the chosen slot (Lines 2, 3 in Algorithm 1). We store the observed pick-ups and engagements of the beneficiaries in a matrix $\mathbf{M}$ (Line 4 in Algorithm 1). At the end of the exploration phase, we use the observed data $\mathbf{M}$ to estimate the unknown pick-up/engagement matrix. To do this, we propose a novel bagging-based use of the matrix completion oracle $\mathcal{O}$ (Algorithm 3) that we detail below. Subsequently, in the exploitation phase, we use the estimated matrix to find the most preferred time slot for each beneficiary and commit to that slot for the remaining $\mathsf{T} - \mathsf{T}_{\text{explore}}$ rounds (Lines 7-9 in Algorithm 1).

Robust Median Estimates using Bagging (Algorithm 3): Naively performing MC on the observed data ($\mathbf{M}$) to infer $\mathbf{P}$ (or $\mathbf{E}$) leads to estimates with high variance. To address this, we infer $\mathbf{P}$ (or $\mathbf{E}$) using multiple sub-samples of the data, and aggregate the estimates to produce a final estimate with reduced variance [8]. Specifically, for each beneficiary u, we consider K small groups of beneficiaries, i.e. $\mathcal{U}_1^u, \mathcal{U}_2^u \ldots \mathcal{U}_K^u$ where $u \in \mathcal{U}_i^u \subset [M]$ (each group contains u). These groups are chosen randomly such that $|\mathcal{U}_i^u| > N$. Suppose $\Omega \subset [M] \times [N]$ is the set of entries for which the pickup (or engagement) data is available in matrix $\mathbf{M}$. Note that $|\Omega|$ is typically much smaller than $\mathsf{N} \times \mathsf{M}$. For each group $\mathcal{U}_i^u$, we use the relevant data obtained from the exploration phase from matrix $\mathbf{M}$, i.e. $\tilde{\Omega}_i = \Omega \cap (\mathcal{U}_i^u \times [N])$ and apply the optimization routine in (3) with $\mathbf{M}_{\mathcal{U}_i^u,[N]}$. We obtain the completed sub-matrix $\widehat{\mathbf{Z}}_{\mathcal{U}_i^u,[N]}$ by using (3). Thus, for each beneficiary u, we obtain K estimates of it's pickup

Algorithm 1 Greedy MC Algorithm for maximizing engagement

Require: exploration rounds $\mathsf{T}_{\text{explore}}$.

Initialize $\Omega \leftarrow \emptyset$, $\mathbf{M} \in \mathbb{R}^{M \times N}$.
for $t = 1, 2, \ldots, \mathsf{T}_{\text{explore}}$ **do**
  For each u in $[\mathsf{M}]$, randomly sample a slot $\rho_u(t)$ and place a call. Let the indicator of engagement be $B_t(u)$
  If $(u, \rho_u(t)) \notin \Omega$, then $\Omega \leftarrow \Omega \cup (u, \rho_u(t))$, $\mathbf{M}(u, \rho_u(t)) \leftarrow B_t(u)$. If $(u, \rho_u(t)) \in \Omega$, then $\mathbf{M}(u, \rho_u(t)) \leftarrow \frac{B_t(u)}{t} + (1 - \frac{1}{t})\mathbf{M}(u, \rho_u(t))$
end for
For each beneficiary u in $[\mathsf{M}]$, estimate the engagement rates for all the slots, $\mathbf{E}_u \leftarrow \text{MC\_RME}(u, M, \Omega)$.
for each of remaining rounds **do**
  For each beneficiary u, choose the slot from vector $\mathbf{E}_u$ with highest estimated probability and make a call.
end for

Algorithm 2 Phased MC Algorithm for maximizing engagement

Require: Phase length Δ, temperature parameter β.

1: Initialize row stochastic matrix $\mathbf{Q} \in \mathbb{R}^{\mathsf{M} \times \mathsf{N}}$ with $\mathbf{Q}_{ij} = \mathsf{N}^{-1}$ for all $(i, j) \in [\mathsf{M}] \times [\mathsf{N}]$. Initialize $\Omega \leftarrow \emptyset$, $\mathbf{M} \in \mathbb{R}^{M \times N}$.
2: **for** phase $= 1, 2, \ldots, \lceil \mathsf{T}/\Delta \rceil$ **do**
3:   **for** $t = 1, 2, \ldots, \min(\Delta, \mathsf{T} - \text{phase} \cdot \Delta)$ **do**
4:     For each u in $[\mathsf{M}]$, sample a slot $\rho_u(t) \sim \mathbf{Q}_u$ and place a call. Let the indicator of engagement be $B_t(u)$.
5:     If $(u, \rho_u(t)) \notin \Omega$, then $\Omega \leftarrow \Omega \cup (u, \rho_u(t))$, $\mathbf{M}(u, \rho_u(t)) \leftarrow B_t(u)$. If $(u, \rho_u(t)) \in \Omega$, then $\mathbf{M}(u, \rho_u(t)) \leftarrow \frac{B_t(u)}{t} + (1 - \frac{1}{t})\mathbf{M}(u, \rho_u(t))$
6:   **end for**
7:   For each beneficiary u in $[\mathsf{M}]$, estimate the engagement rates for all the slots, $\mathbf{E}_u \leftarrow \text{MC\_RME}(u, M, \Omega)$.
8:   For each beneficiary $u \in [\mathsf{M}]$ and each slot $j \in [\mathsf{N}]$, update $\mathbf{Q}_{uj} \leftarrow \exp(\beta \mathbf{E}_{uj}) \Big(\sum_{j' \in [\mathsf{N}]} \exp(\beta \mathbf{E}_{uj'}) \Big)^{-1}$.
9: **end for**

Algorithm 3 Matrix Completion with Robust Median Estimates: MC_RME(u, M, Ω)

Require: User u, Observed Data $\mathbf{M} \in \mathbb{R}^{\mathsf{M} \times \mathsf{N}}$, set of observed entries Ω, Low Rank MC Oracle $\mathcal{O}$.

Construct K groups $\mathcal{U}_1^u, \mathcal{U}_2^u \ldots \mathcal{U}_K^u$ of $\mathsf{N}' > \mathsf{N}$ beneficiaries, each comprising u. In each group, the beneficiaries other than u are sampled uniformly at random without replacement.
For each group of beneficiaries $\mathcal{U}_i^u$, invoke the MC completion oracle $\mathcal{O}$, i.e. solving the optimization problem in (3), using $\tilde{\Omega}_i = \Omega \cap \mathcal{U}_i^u \times [N]$ and observed data $\mathbf{M}_{\mathcal{U}_i^u, [N]}$, to compute an estimate of $\widehat{\mathbf{Z}}^i_{\mathcal{U}_i^u, [\mathsf{N}]}$ (sub-matrix corresponding to rows in $\mathcal{U}_i^u$ and set of slots $[\mathsf{N}]$).
Construct final estimate of $\mathbf{E}_u$ by computing entry-wise median of the K estimates of row u, i.e. $\mathbf{E}_u = \text{Median}(\{\widehat{\mathbf{Z}}^i_{u,:}\}_{i=1}^K)$
Return $\mathbf{E}_u$

(or engagement) probabilities at each slot, and compute the entry-wise median of the K estimates and use it as the final estimate for that beneficiary. This procedure helps reduce the variance in our estimates, and makes it robust to outliers [11, 25, 33].

3.1.2 Phased Matrix Completion (Algorithm 2). Although the Greedy MC algorithm is conceptually quite simple, it has one main drawback: it needs a lengthy exploration phase in the beginning to get a good estimate of $\mathbf{P}, \mathbf{E}$ (see Figure 12). However, this can hurt the user experience, and can even drive beneficiaries away from the program, as slots are chosen randomly. To address this limitation, we propose an alternate algorithm called Phased Matrix Completion (MC) which reduces the initial exploration period.

The Phased MC algorithm partitions the T rounds into equally sized $\lceil \mathsf{T}/\Delta \rceil$ phases of length Δ (Lines 2, 3 in Algorithm 2). The first phase is similar to the exploration phase of Greedy MC; *i.e.,* in each round of this phase, for each beneficiary, a slot is chosen uniformly at random and a call is placed. At the end of each phase, we rely on the data collected so far ($\mathbf{M}$) to estimate $\mathbf{P}$ (or $\mathbf{E}$) by invoking the robust matrix completion subroutine (Algorithm 3). The key novelty in our algorithm is in the kind of exploration we perform in each phase. Within each phase, for each beneficiary, we sample slots with probability proportional to exponential of the estimated pick-up/engagement rate of the slot (modulo a scaling factor β). β provides a trade-off between exploration and exploitation, with larger values favouring exploitation and smaller values favouring exploration. This approach is also known as Boltzmann exploration, and has been recently studied in the context of classical multi-armed bandits [12]. We note that Δ in this algorithm is smaller than the exploration phase of Greedy MC. This helps us pick meaningful slots after the end of first phase itself (see Figures 13a, 13b). However, in contrast to Greedy MC, the Phased MC algorithm is computationally more intensive as it needs to compute $\lfloor \mathsf{T}/\Delta \rfloor$ estimates - one after each phase - of $\mathbf{P}, \mathbf{E}$ (see Appendix A).

4 EXPERIMENTS

Dataset: We obtained an anonymized call log dataset from our NGO partner ARMMAN. This data was collected over a period of one year, and has $M = 200K$ beneficiaries, $N = 7$ time slots (across 8am to 8pm) at which the calls were made. Using this data, we first constructed "ground truth" matrices $\mathbf{P}, \mathbf{E}$ for simulation. Specifically, we estimated the pick-up (engagement) probability for each (beneficiary, slot) tuple as the ratio of number of times the beneficiary picked-up (engaged) to the number of calls placed in that slot. In all our experiments, we ensure that the algorithms don't have access to the matrices $\mathbf{P}, \mathbf{E}$. We use these matrices solely to simulate binary observations (pick-up, engagement). Finally, we note that the constructed matrices $\mathbf{P}, \mathbf{E}$ are completely filled (a small fraction of entries are missing in the ground truth matrix but they were imputed using ad-hoc techniques that are completely agnostic to the algorithm). As stated in the introduction, we use the following two metrics to compare various algorithms: (a) $\mathsf{Reg}_{\text{pick-up}}(\mathsf{T})$, $\mathsf{Reg}_{\text{engage}}(\mathsf{T})$ which measure the expected pick-up, engagement rates of an algorithm, and (b) number of retries before a successful call. The major takeaways from our experiments with the retries constraint (Section 4.2) that model the practical Kilikari set-up are the following:

- The MC based algorithmic framework (Greedy MC and Phased MC) obtain significant reduction in regret over the non-collaborative UCB policy (> 27%).
- Our MC based algorithms obtain a significant reduction in the number of average retries for 7 slots. The reduction over random policies and UCB policy is > 30% and > 9% respectively for several groups of beneficiaries.
- On extending our constrained setting to 14 time slots by taking into account, the weekday-weekend flag, the reduction in average retries for MC based algorithms become even more pronounced and goes up to > 45% over random and > 25% over UCB policies.

4.1 Online Collaborative Learning

In this section, we demonstrate the efficacy of collaborative bandit algorithms in identifying appropriate time-slots for beneficiaries. We first describe our experimental setting. We subsample $M = 1000$ beneficiaries uniformly at random from the $200K$ beneficiaries, and consider $T = 50$ rounds. At each round, for each of the M beneficiaries, a sequential algorithm chooses a time slot to call the beneficiary based on the data obtained in previous rounds. We simulate calls, pick-ups and engagements using the "ground truth" $\mathbf{P}, \mathbf{E}$ matrices as follows: the beneficiary u picks up the call made in the chosen time-slot $\rho_u(t)$ at round t with probability $\mathbf{P}_{u\rho_u(t)}$ and engages with probability $\mathbf{E}_{u\rho_u(t)}[\mathbf{P}_{u\rho_u(t)}]^{-1}$ conditioned on the pick-up. We compare the following algorithms: (a) UCB (Upper Confidence Bound) which treats each beneficiary independently[1], (b) Greedy Matrix Completion with $\mathsf{T}_{\text{explore}}$ set to 5, 10, and (c) Phased Matrix Completion with $\Delta = 5$. We repeat this experiment 15 times with different subset of beneficiaries. We used grid search to select the exploration hyper-parameter in UCB, $\mathsf{T}_{\text{explore}}$ in Greedy MC and Δ in Phased MC.

In addition to regret, for a more intuitive evaluation, we also compare the algorithms in terms of the average rank of the chosen time-slot at each round. Note that for each beneficiary, we only need to learn the right ranking of the available time-slots based

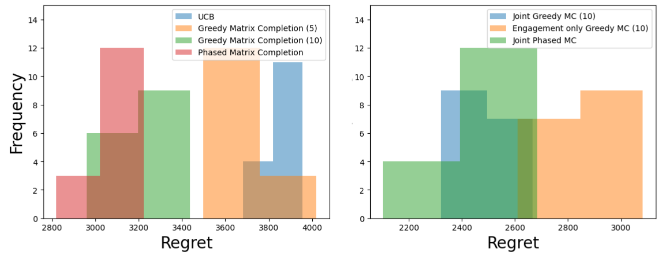


Table 1: (Left Figure) Histogram of regret $\mathsf{Reg}_{\textbf{pick-up}}(\mathsf{T})$ (for $\mathsf{T} = 50$ rounds) across 15 simulation runs of the 3 algorithms - 1) UCB 2) Greedy MC with 5, 10 exploration rounds 3) Phased MC. (Right Figure) Histogram of regret $\mathsf{Reg}_{\textbf{engage}}(\mathsf{T})$ for the same setup - 1) Engagement data-only Greedy MC with 10 exploration rounds 2) Joint Greedy MC with 10 exploration rounds 3) Joint Phased MC. Note that the regret accrued by MC algorithms are significantly smaller than UCB.

on their preferences. Based on this observation, at each round, we will also consider the position (zero-indexed) of the time slot (position among time slots sorted in descending order of preference - pickup/engagement probability) chosen by the algorithm for a particular beneficiary and subsequently its average across all beneficiaries. For example, at the 10^{th} round, an average position of 1.5 implies that the time slot chosen by the algorithm at the 10^{th} round is roughly the 2.5^{th} best time slot for beneficiaries. Naturally, the average time slot position is expected to decrease with rounds.

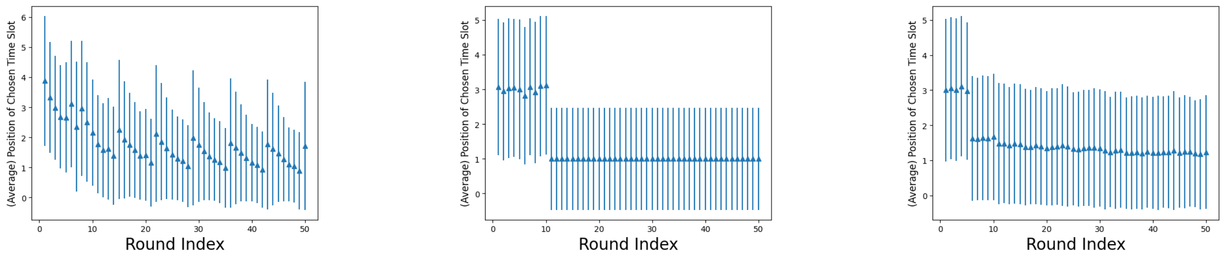
4.1.1 Pick-up data. In this sub-section, we will focus on the pick-up matrix $\mathbf{P}$. Here, all the algorithms try to minimize the regret related to pick-ups $\mathsf{Reg}_{\text{pick-up}}(\mathsf{T})$.

The left plot of Table 1 presents a histogram of regret of the three algorithms described above, across 15 simulation runs. It is clear from the figure that the Phased MC algorithm and the Greedy MC algorithm (with 10 exploration rounds) significantly outperform UCB (> 20% improvement in regret) and have the best performance overall. Finally, we note that the purely randomized policy that is currently implemented by the NGO has a regret more than 10000.

In order to understand intuitively the reason behind the improved performance of the MC algorithms, note that the top 3 eigenvalues of the gram matrix of $\mathbf{P}$ are $[182469.5, 24910.2, 7026.8]$. Clearly the first eigen-value is approximately 6 times the second one which in turn is approximately 3 times the third eigen-value. Thus, we can conclude that the pickup estimate matrix $\mathbf{P}$, despite being an incredibly tall matrix with 7 columns can be well approximated by a rank-2 matrix. This, in turn, implies that a significant amount of information is shared across the beneficiaries. This is the crucial structural prior exploited by MC algorithms. In contrast, UCB is implemented separately for each beneficiary and therefore cannot take advantage of the shared information across beneficiaries.

In Figure 1 we highlighted the average ranking (across beneficiaries) of the chosen time-slot by various algorithms. It is clear from the mean-variance plots that Greedy MC (with 10 exploration rounds) and Phased MC have significantly improved choice of slots as the learning progresses. We remark that there is a periodic structure in the choice of slots for the UCB algorithm - this stems from the algorithmic design itself where the constructed confidence interval is large enough for several beneficiaries who sequentially

[1] In the absence of user demographic features, UCB/TS are the best non-collaborative baselines that attempt to intelligently elicit preferred timeslot info from each user.



(a) UCB **(b) Greedy MC with 10 exploration rounds** **(c) Phased MC**

Figure 1: (Pick-up) Figure shows the average rank/position of time-slot chosen, as a function of rounds. Each vertical bar represents the mean, variance of the rank. Note that the MC algorithms choose significantly better time-slots on average, as the algorithms progress.

go through all slots in a round-robin fashion. In the Greedy MC, in the initial exploration component, slots are chosen uniformly at random - thus the mean is close to 3. In the exploitation component, the greedy MC algorithm commits to a fixed time-slot for every beneficiary and therefore, vertical bars are identical across rounds in exploitation component. For Phased MC, note that the algorithm continues to improve its choice of slot gradually for all beneficiaries.

We also perform a similar set of experiments on the engagement data, where the goal is to minimize the engagement related regret $\text{Reg}_{\text{engage}}(\mathsf{T})$. We obtain similar trends as in the pickup setting - the relevant results are provided in Appendix D.1.

4.1.2 Combined Pick-up and Engagement data. Here, we aim to minimize the engagement related regret $\text{Reg}_{\text{engage}}(\mathsf{T})$. However, in contrast to experiments in Appendix D.1 where we relied solely on engagement data, here we try to exploit both pick-up and engagement data to improve $\text{Reg}_{\text{engage}}(\mathsf{T})$. While pick-up rate is only a noisy signal of engagement rate, it is a much denser signal than engagement. Consequently, we use it to augment the engagement data to improve the performance of MC algorithms. Intuitively, there can be several scenarios when the beneficiary had picked up but didn't engage (listened to less than 25%) due to several reasons (inconvenient time or call picked up by family member or lack of interest). In other words, even if the beneficiary picks up but does not engage, it might lead to some information about the engagement itself. The precise goal is to reduce $\text{Reg}_{\text{engage}}(\mathsf{T})$ by using pickup and engagement information jointly over just using engagement data. This is a challenging problem itself since it is non-trivial on how to model the interaction between pick-up and engagement.

Due to above reasons, extending baseline non-collaborative algorithms such as UCB to model interaction and pickups seems complicated - it would necessarily entail making a certain set of assumptions. However, the MC framework provides a very convenient and elegant solution - the main idea is to jointly estimate both pick-up and engagement matrices by combining all observations. More precisely, consider the ground truth matrix $\mathbf{R}$ to be a concatenation of the pickup and engagement matrices $\mathbf{P}, \mathbf{E}$ respectively - as usual, $\mathbf{R}$ is unknown to the algorithm. As before, in a single simulation run, we have $\mathsf{M} = 1000$ randomly sampled users. At each round, for each beneficiary, a slot is chosen to make a call - subsequently two binary observations are made corresponding to a pick-up and and engagement conditioned on a pick-up. Based on these noisy binary observations, we impute all missing entries of $\mathbf{R}$ jointly whenever we invoke the offline low rank MC algorithms. We compare the regret of the following algorithms: (a) Greedy MC algorithm with 10 exploration rounds that minimizes $\text{Reg}_{\text{engage}}(\mathsf{T})$ based solely on engagement data (b) Joint greedy MC algorithm with 10 exploration rounds, and (c) Joint phased MC algorithm. Note that (b), (c) perform MC on the joint pick-up and engagement data. The right figure of Table 1 presents the histogram of regret of various algorithms over 15 simulation runs. Notice that our Joint MC approaches achieve 10% improvement over the greedy MC algorithm which only relies on engagement data.

4.2 Handling Retry Constraints

In practice, the NGOs are usually faced with resource constraints and have an limit on the number of calls they can make to the beneficiaries. To make our algorithms deployable in practice, we now modify them to handle two such constraints that arise in the context of Kilkari: (a) at most 9 attempts can be made to reach a beneficiary via calls in each week (b) if a call is successful for a particular beneficiary, then no other attempts are made in that week to reach out to that beneficiary. In this setting, an important metric to evaluate our algorithms is to demonstrate improvement in the average retries needed before a successful call to the beneficiary. As a service provider, if the average retries is reduced, there is a significant increase in the capacity. This extra capacity can be used to place further calls to low-engagement beneficiaries, as well as potentially increase enrolments into the program which are currently limited due to the infrastructural constraints.

As before, in each run of the simulation, we sample $\mathsf{M} = 1000$ beneficiaries from the $200k$ beneficiaries uniformly at random and simulate calls, pick-ups and engagements. In this setting, we consider $\mathsf{T} = 270$ rounds - each week comprises of 9 rounds and thus, we simulate our experiment over 30 weeks of data. At each particular round in a week, we only simulate a call and pickup (or engagement) for those beneficiaries who have not picked up (or engaged) in any of the previous calls made in that week. Since we do not call every beneficiary in each round, we define a modified version of the usual regret $\text{Reg}^{\text{new}}_{\text{pick-up}}(\mathsf{T})$, $\text{Reg}^{\text{new}}_{\text{engage}}(\mathsf{T})$ for pick-up and engagement respectively. In these definitions, for each beneficiary, we only consider the rounds when calls are placed to them. More precisely, let $\mathcal{T}_u \subset [\mathsf{T}]$ be the set of rounds when calls are made to beneficiary u. We define the regret in this setting as

$$\text{Reg}^{\text{new}}_{\text{pick-up}}(\mathsf{T}) = \sum_{u \in [\mathsf{M}]} \Big(|\mathcal{T}_u| \, \mathbf{P}_{u\pi_p(u)} - \mathbb{E}_{\mathcal{A}} \sum_{t \in \mathcal{T}_u} \mathbf{P}_{u\rho_u(t)} \Big)$$

$$\mathrm{Reg}^{\mathrm{new}}_{\mathrm{engage}}(\mathsf{T}) = \sum_{u\in[\mathsf{M}]} \Big(|\mathcal{T}_u|\, \mathbf{E}_{u\pi_e(u)} - \mathbb{E}_{\mathcal{A}} \sum_{t\in\mathcal{T}_u} \mathbf{R}_{u\rho_u(t)} \Big).$$

With the above set-up, we again compare the regret and the average retries of the 3 aforementioned algorithm - (a) UCB (Upper Confidence Bound) implemented separately for each beneficiary (b) Greedy MC with exploration periods of 27, 45 rounds and (c) Phased MC with $\Delta = 27$. We implement our algorithms by simulating calls, pickups and engagement using the ground truth matrices related to pick-up ($\mathbf{P}$), engagement ($\mathbf{E}$) respectively.

4.2.1 Pickup Data: In this sub-section, we will again focus on the ground truth matrix $\mathbf{P}$ to simulate call pick-ups. For each beneficiary, we have the following template - 1) the 270 rounds are partitioned into 30 groups (representing weeks) of 9 rounds each 2) In each group of rounds, our designed algorithm chooses a slot to call the beneficiary until they have picked-up - pickups are simulated by entries of $\mathbf{P}$ as is usual 3) once the beneficiary picks up, no more calls are placed to that beneficiary in remaining rounds in that group 4) The algorithm restarts making calls in the subsequent week to the beneficiary.

Our simulation results (across 15 simulation runs) with pick-up data are summarized in Figures 2a, 2b and 2c. Figures 2a compares the regret for each simulation run across 15 runs. As in previous experiments, a random policy accrues a 10 times higher regret of more than 50000 in each simulation run. In Figure 2a, it is clear that there is a significant reduction in regret of more than 33% ($\mathrm{Reg}^{\mathrm{new}}_{\mathrm{pick\text{-}up}}(\mathsf{T})$ for $\mathsf{T} = 270$) by the Greedy and Phased algorithms in MC framework over non-collaborative UCB algorithm. In turn, this also translates into a reduction of > 5% in the average number of call retries for pickup (average across users and rounds), for MC based algorithms over UCB. In Figures 2b and 2c, we dive deeper into the analysis of average retries.

Note that in our dataset, there are several beneficiaries who are low-pickup - in other words, no matter the slot that is recommended to these beneficiaries, they are unlikely to pickup and engage. For these beneficiaries, the choice of algorithm is almost irrelevant especially when there is a cap of 9 retries per week for each beneficiary (shown by red horizontal lines in the figures). To understand this better, we study the reduction in average retries of beneficiaries by stratifying them according to the maximum pick-up ground truth probabilities. We have 6 bins (partitioning the probability range) that comprise of the intervals $[0, 0.1]$, $[0.1, 0.2]$ and the interval $[0.2, 1]$ partitioned into 4 equal intervals. More precisely, the bin $[a, b]$ comprises all beneficiaries each of whom satisfies the following condition - the maximum probability of pick-up assigned to some slot for each of the aforementioned beneficiaries lies in the interval $[a, b]$. For each bin, we report 1) the average number of beneficiaries in that bin 2) average reduction in percentage of retries as compared to UCB policy 3) average reduction in percentage of retries as compared to a random policy - here the average is computed across all 15 simulation runs. The non-uniform splitting of bins is to highlight low pick-up users (bin $[0, 0.1]$) in particular - here the choice of algorithms is almost inconsequential as beneficiaries rarely pick-up. Yet, even for the aforementioned bin, greedy MC leads to a reduction of > 4% over random policy and > 1.5% over UCB.

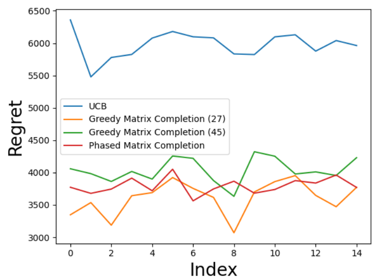
Clearly, in Figure 2b, the greedy MC with 27 exploration rounds leads to more significant reduction in average retries when the maximum ground truth pick-up probabilities is neither large nor small - we have 1) 14.49% and > 35% reduction in average retries over UCB and random policy respectively for the 195 beneficiaries (on average) that lie in the bin $[0.4.0.6]$ 2) > 7% and > 35% reduction in average retries over UCB and random policy respectively for 332 beneficiaries (on average) in the two adjacent bins. This in particular highlights the efficacy of our algorithm. Note that for very low pickup beneficiaries and very high pick-up beneficiaries, UCB and Greedy MC have similar performance - however, in the latter case, there is a > 37% reduction in average retries.

Next, for Figure 2c, we consider a more complex setting with 14 time-slots (the 7 time slots considered previously along with weekday/weekend flag). With this granular definition of time slots, the beneficiaries have a significant amount of missing data. Out of approximately $200k$ beneficiaries in the actual data-set, we focused on $\approx 23k$ beneficiaries who had ground truth data for at least 10 slots out of the 14. For each of these $\approx 23k$ beneficiaries, we impute the missing data by simply taking the average of the available probabilities for the $\geq$ 10 slots of the concerned. Fig. 2c report average retries until pickup/engagement with retries constraint for 14 slots corresponding to 7 slots of the day combined with a weekday/weekend flag. We find upto > 45% benefit in retries over random policy and upto > 30% benefit in retries over UCB policy.

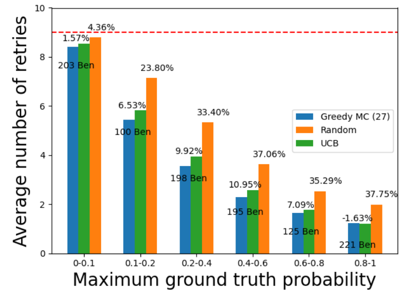
Finally, if we consider convergence within 0.15 of the true highest probability slot, Greedy MC converges for 93% beneficiaries after 27 rounds (by design), while UCB converges for 85.5% beneficiaries within 270 rounds (and 62% in 27 rounds), and fails to converge for 15.5% beneficiaries even after 270 rounds. This highlights the quick convergence of the algorithm for majority of the population enabling the NGO to act fast to avoid dropoffs from the program and boost engagement.

4.2.2 Engagement Data. We now focus on the ground truth matrix $\mathbf{E}$ to simulate call engagements. We have the same setting as in the pick-up case. In a particular simulation run, for each beneficiary among 1000 randomly sampled beneficiaries, 1) the $\mathsf{T} = 270$ rounds are partitioned into 30 groups of 9 rounds each 2) In each group of rounds, our designed algorithm chooses a slot to call the beneficiary until they have engaged - simulated by entries of $\mathbf{E}$ 3) once the beneficiary engages, the algorithm no longer places calls to that beneficiary in that week and restarts next week.

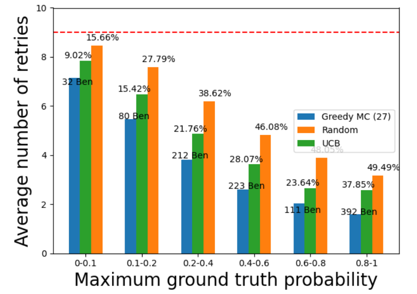
Our simulations with the engagement data are summarized in Figures 2d, 2e and 8b with analogous conclusions to the pickup setting. As before, we can conclude from the first figure that there is a significant reduction in regret (> 27%) accrued by MC algorithms over UCB. This, in turn, leads to a reduction in the number of average call retries for beneficiary engagement. This reduction is lower than that obtained for the pickup setting but it stems from the presence of a significantly larger fraction of low-engagement beneficiaries as compared to low pick-up beneficiaries. In Figure 2e, we stratify the users according to the maximum engagement probability for some slot. As before, we partition the range of engagement probabilities $[0, 1]$ into 6 disjoint intervals - $[0, 0.1]$, $[0.1, 0.2]$ and $[0.2, 1]$ split into 4 intervals. Notice that the reduction in average retries over the random policy is significantly large and is more than



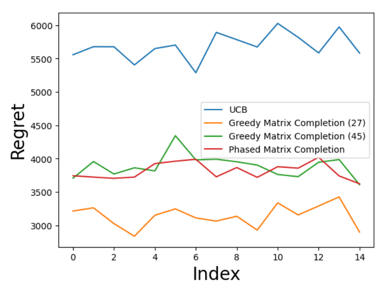
(a) Comparison of $\text{Reg}^{\textbf{new}}_{\textbf{pick-up}}(\text{T})$



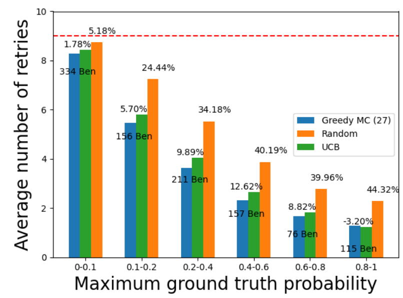
(b) Retries across max probability bins



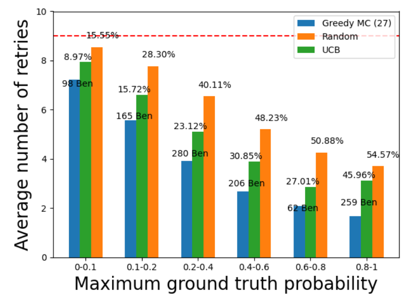
(c) Retries across max probability bins with weekday/weekend flags



(d) Comparison of $\text{Reg}^{\textbf{new}}_{\textbf{engage}}(\text{T})$



(e) Retries across max-probability bins



(f) Retries across max probability bins with weekday/weekend flags

Figure 2: Our experiments with retries constraint on a weekly basis. The top row shows figures corresponding to our results for pick-up (bottom row for engagement) with the retries constraint. In the first column, we compare the regret for pick-up (engagement),i.e. $\text{Reg}^{\textbf{new}}_{\textbf{pick-up}}(\text{T})$ ($\text{Reg}^{\textbf{new}}_{\textbf{engage}}(\text{T})$), for $\text{T} = 270$ rounds across 15 simulation runs for 4 distinct algorithms. 1) UCB 2) Greedy MC with $27, 45$ exploration rounds 3) Phased MC. Clearly, the MC based algorithms lead to $> 27\%$ reduction in regret over non-collaborative algorithms. Figures in second column report reduction in average retries of MC based algorithms over baselines with beneficiaries slotted into bins. Bin $[a, b]$ has beneficiaries whose true max probability of pick-up (engagement) lies in $[a, b]$. Again the MC based algorithms show significant reduction in average retries over random policy ($> 30\%$ in some cases) and over UCB policy ($> 9\%$ in some cases). Fig. 2c and Fig. 2f report average retries until pickup/engagement with retries constraint for 14 slots corresponding to 7 slots of the day combined with a weekday/weekend flag. We find upto $> 45\%$ benefit in retries over random policy and upto $> 30\%$ benefit in retries over UCB policy.

40% for certain bins. Similarly, the reduction in average retries over UCB also goes to $> 12\%$ in certain bins. Again the objective of non-uniform splitting is to highlight the low-engagement beneficiaries. Figure 2e show that on average 33.4% of beneficiaries are extremely low engagement with maximum engagement probability for some slot to be between $0 - 0.1$. Even for these low-engagement beneficiaries, Greedy MC gets a $> 5\%$ and 1.78% reduction in average retries over random policy and UCB respectively. The improvement becomes more pronounced with higher-engagement users. Similar to the pick-up setting, on generalization our experiments to the case of 14 slots, the improvement in average retries again become significantly pronounced - these results are summarized in Fig. 2f.

Combined Pickup and Engagement data: We repeated the greedy version of our algorithms with the combined pick-up and engagement data to minimize the regret for engagement. However, in this setting, we find nominal gains on combining - over 15 simulation runs, the greedy algorithm on concatenated pickup and engagement data with 27 exploration rounds accrue average $\text{Reg}^{\text{new}}_{\text{engage}}(270)$ of 3072.69 while the greedy algorithm on sole engagement data with 27 exploration rounds accrue average $\text{Reg}^{\text{new}}_{\text{engage}}(270)$ of 3182.25.

5 CONCLUSION

We presented two methods inspired by collaborative bandits that exploit the low-rank structure of the problem to infer optimal time slots to boost listenership with the largest maternal mHealth program in the world. Additionally, we strengthen our models by combining pickup and engagement signals. We conducted multiple experiments with real-world data obtained from the NGO ARMMAN to show both the methods outperform the current baseline deployed by the NGO as well as a non collaborative approach (UCB). Particularly, for 7 time slot problem, the average number of retries needed to reach a person drastically reduces by 30% and 9% compared to random and UCB for 42% beneficiaries, and over 45% and 25% reduction for 14 time slots respectively (for 76% beneficiaries), saving critical bandwidth for the program to enable reaching out to more beneficiaries. For the 14 time slots case, we show that one can optimistically reach 56% more beneficiaries with our MC based approach when compared to UCB based non collaborative policies when resources for both are equalized. Additionally, the proposed methods converge to within 0.15 of the best time slot for 93% beneficiaries in 3 weeks, enabling the NGO to act fast and hence retain beneficiaries in the program as well as boost their engagement.

6 ETHICS STATEMENT

The analysis presented in this paper falls into the category of secondary analysis of the anonymized dataset obtained from our NGO partner ARMMAN. There is no demographic or personally identifiable information available. We only use previously collected listenership trajectories of beneficiaries participating in the Kilkari program to train the predictive model and evaluate it's performance. All the data collected through the program is owned by the NGO and only the NGO is allowed to share data.

Bias and fairness. Prior studies on Kilkari [27] point out that exposure to Kilkari helps improve health behaviors among the most marginalised, and that the more marginalised population benefits from higher number of retries in Kilkari calls. While we don't have access to demographic data, we do hope for the proposed method to potentially help reduce such inequities by improving listenership of Kilkari messages particularly amongst low listeners.

Path to deployment. The proposed method is intended to be deployed at a national scale in India. With that goal in mind, the next step will involve a randomized control trial in one state in India to validate the true usefulness of the method in the field. With Kilkari being operational in 19 states in India, the model can then be deployed gradually across the different regions. Naturally a deployment at this scale and diversity of beneficiaries may reveal new challenges, such as regional differences in listenership patterns, bandwidth limitations. Most importantly, though, all of the steps will be done in close collaboration with our partner ARMMAN; with ARMMAN ultimately in charge of the actual deployment.

REFERENCES

[1] Shipra Agrawal and Navin Goyal. 2012. Analysis of thompson sampling for the multi-armed bandit problem. In *Conference on learning theory*. JMLR Workshop and Conference Proceedings, 39–1.

[2] Shubhada Agrawal, Sandeep Juneja, and Peter Glynn. 2020. Optimal δ-Correct Best-Arm Selection for Heavy-Tailed Distributions. In *Proceedings of the 31st International Conference on Algorithmic Learning Theory (Proceedings of Machine Learning Research, Vol. 117)*. PMLR, 61–110.

[3] ARMMAN. 2023. Kilkari. https://armman.org/kilkari/

[4] Jean-Yves Audibert, Sébastien Bubeck, and Rémi Munos. 2010. Best arm identification in multi-armed bandits.. In *COLT*. 41–53.

[5] Peter Auer, Nicolo Cesa-Bianchi, and Paul Fischer. 2002. Finite-time analysis of the multiarmed bandit problem. *Machine learning* 47 (2002), 235–256.

[6] Jean Juste Harrisson Bashingwa, Diwakar Mohan, Sara Chamberlain, Salil Arora, Jai Mendiratta, Sai Rahul, Vinod Chauhan, Kerry Scott, Neha Shah, Osama Ummer, et al. 2021. Assessing exposure to Kilkari: a big data analysis of a large maternal mobile messaging service across 13 states in India. *BMJ Global Health* 6, Suppl 5 (2021), e005213.

[7] James Bennett, Stan Lanning, et al. 2007. The netflix prize. In *Proceedings of KDD cup and workshop*, Vol. 2007. New York, 35.

[8] Leo Breiman. 1996. Bagging predictors. *Machine learning* 24 (1996), 123–140.

[9] Guy Bresler, Devavrat Shah, and Luis Filipe Voloch. 2016. Collaborative Filtering with Low Regret. In *Proceedings of the 2016 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Science (SIGMETRICS '16)*. ACM, New York, NY, USA, 207–220.

[10] Jean-Philippe Brunet, Pablo Tamayo, Todd R Golub, and Jill P Mesirov. 2004. Metagenes and molecular pattern discovery using matrix factorization. *Proceedings of the national academy of sciences* 101, 12 (2004), 4164–4169.

[11] Peter Buhlmann, Bin Yu, et al. 2002. Analyzing bagging. *Annals of statistics* 30, 4 (2002), 927–961.

[12] Nicolò Cesa-Bianchi, Claudio Gentile, Gábor Lugosi, and Gergely Neu. 2017. Boltzmann exploration done right. *Advances in neural information processing systems* 30 (2017).

[13] Yuxin Chen, Yuejie Chi, Jianqing Fan, Cong Ma, and Yuling Yan. 2019. Noisy matrix completion: Understanding statistical guarantees for convex relaxation via nonconvex optimization. *arXiv preprint arXiv:1902.07698* (2019).

[14] Hamid Dadkhahi and Sahand Negahban. 2018. Alternating Linear Bandits for Online Matrix-Factorization Recommendation. *arXiv preprint arXiv:1810.09401* (2018).

[15] Aurélien Garivier and Emilie Kaufmann. 2016. Optimal best arm identification with fixed confidence. In *Conference on Learning Theory*. PMLR, 998–1027.

[16] Claudio Gentile, Shuai Li, and Giovanni Zappella. 2014. Online clustering of bandits. In *International Conference on Machine Learning*. PMLR, 757–765.

[17] Jacaranda Health. 2023. Jacaranda Health. https://www.jacarandahealth.org/prompts

[18] Prateek Jain, Praneeth Netrapalli, and Sujay Sanghavi. 2013. Low-rank matrix completion using alternating minimization. In *Proceedings of the forty-fifth annual ACM symposium on Theory of computing*. 665–674.

[19] Prateek Jain and Soumyabrata Pal. 2022. Online Low Rank Matrix Completion. *arXiv preprint arXiv:2209.03997* (2022).

[20] Kevin Jamieson, Matthew Malloy, Robert Nowak, and Sébastien Bubeck. 2014. lil' UCB : An Optimal Exploration Algorithm for Multi-Armed Bandits. In *Proceedings of The 27th Conference on Learning Theory (Proceedings of Machine Learning Research, Vol. 35)*, Maria Florina Balcan, Vitaly Feldman, and Csaba Szepesvári (Eds.). PMLR, Barcelona, Spain, 423–439. https://proceedings.mlr.press/v35/jamieson14.html

[21] Zohar Karnin, Tomer Koren, and Oren Somekh. 2013. Almost Optimal Exploration in Multi-Armed Bandits. In *Proceedings of the 30th International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 28)*, Sanjoy Dasgupta and David McAllester (Eds.). PMLR, Atlanta, Georgia, USA, 1238–1246. https://proceedings.mlr.press/v28/karnin13.html

[22] Vladimir Koltchinskii, Karim Lounici, and Alexandre B Tsybakov. 2011. Nuclear-norm penalization and optimal rates for noisy low-rank matrix completion. (2011).

[23] Arshika Lalan, Shresth Verma, Kumar Madhu Sudan, Amrita Mahale, Aparna Hegde, Milind Tambe, and Aparna Taneja. 2023. Analyzing and Predicting Low-Listenership Trends in a Large-Scale Mobile Health Program: A Preliminary Investigation.

[24] Tor Lattimore and Csaba Szepesvári. 2020. *Bandit algorithms*. Cambridge University Press.

[25] Guillaume Lecué and Matthieu Lerasle. 2020. Robust machine learning by median-of-means: theory and practice. *Annals of Statistics* (2020).

[26] Aditya Mate, Lovish Madaan, Aparna Taneja, Neha Madhiwalla, Shresth Verma, Gargi Singh, Aparna Hegde, Pradeep Varakantham, and Milind Tambe. 2022. Field study in deploying restless multi-armed bandits: Assisting non-profits in improving maternal and child health. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 36. 12017–12025.

[27] Diwakar Mohan, Kerry Scott, Neha Shah, Jean Juste Harrisson Bashingwa, Arpita Chakraborty, Osama Ummer, Anna Godfrey, Priyanka Dutt, Sara Chamberlain, and Amnesty Elizabeth LeFevre. 2021. Can health information through mobile phones close the divide in health behaviours among the marginalised? An equity analysis of Kilkari in Madhya Pradesh, India. *BMJ Global Health* 6, Suppl 5 (2021), e005512.

[28] MomConnect. 2023. MomConnect. https://www.health.gov.za/momconnect/

[29] Vineet Nair, Kritika Prakash, Michael Wilbur, Aparna Taneja, Corinne Namblard, Oyindamola Adeyemo, Abhishek Dubey, Abiodun Adereni, Milind Tambe, and Ayan Mukhopadhyay. 2022. ADVISER: AI-Driven Vaccination Intervention Optimiser for Increasing Vaccine Uptake in Nigeria. https://arxiv.org/pdf/2204.13663.pdf

[30] World Health Organization. 2023. Maternal Mortaility. https://www.who.int/news-room/fact-sheets/detail/maternal-mortality

[31] World Health Organization. 2023. Sustainable Development Goals. https://www.who.int/data/gho/data/themes/topics/sdg-target-3-1-maternal-mortality

[32] Soumyabrata Pal, Arun Sai Suggala, Karthikeyan Shanmugam, and Prateek Jain. 2023. Optimal Algorithms for Latent Bandits with Cluster Structure. *arXiv preprint arXiv:2301.07040* (2023).

[33] Peter J Rousseeuw and Mia Hubert. 2011. Robust statistics for outlier detection. *Wiley interdisciplinary reviews: Data mining and knowledge discovery* 1, 1 (2011), 73–79.

[34] Sanket Shah, Shresth Verma, Amrita Mahale, Kumar Madhu Sudan, Aparna Hegde, Aparna Taneja, and Milind Tambe. [n.d.]. Preliminary Results in Low-Listenership Prediction in One of the Largest Mobile Health Programs in the World. ([n. d.]).

[35] Aleksandrs Slivkins et al. 2019. Introduction to multi-armed bandits. *Foundations and Trends® in Machine Learning* 12, 1-2 (2019), 1–286.

[36] William R Thompson. 1933. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika* 25, 3-4 (1933), 285–294.

[37] Shresth Verma, Gargi Singh, Aditya S. Mate, Paritosh Verma, Sruthi Gorantala, Neha Madhiwalla, Aparna Hegde, Divy Hasmukhbhai Thakkar, Manish Jain, Milind Shashikant Tambe, and Aparna Taneja. 2023. Deployed SAHELI: Field Optimization of Intelligent RMAB for Maternal and Child Care. In *Innovative Applications of Artificial Intelligence (IAAI)*.

[38] Yiming Xu, Akil Narayan, Hoang Tran, and Clayton G. Webster. 2021. Analysis of the ratio of l1 and l2 norms in compressed sensing. *Applied and Computational Harmonic Analysis* 55 (2021), 486–511. https://doi.org/10.1016/j.acha.2021.06.006