Minimizing Entropy for Crowdsourcing with Combinatorial Multi-Armed Bandit

Yiwen Song[†], Haiming Jin[*‡]
[†]Department of Electronic Engineering, Shanghai Jiao Tong University, Shanghai, China
[‡]John Hopcroft Center for Computer Science, Shanghai Jiao Tong University, Shanghai, China
Email: {gavinsyw, jinhaiming}@sjtu.edu.cn

***Abstract*—Nowadays, crowdsourcing has become an increasingly popular paradigm for large-scale data collection, annotation, and classification. Today's rapid growth of crowdsourcing platforms calls for effective worker selection mechanisms, which oftentimes have to operate with *a priori* unknown worker reliability. We discover that the *empirical entropy* of workers' results, which measures the uncertainty in the final aggregated results, naturally becomes a suitable metric to evaluate the outcome of crowdsourcing tasks. Therefore, this paper designs a worker selection mechanism that minimizes the empirical entropy of the results submitted by participating workers. Specifically, we formulate worker selection under sequentially arriving tasks as a *combinatorial multi-armed bandit* problem, which treats each worker as an arm, and aims at learning the best combination of arms that minimize the cumulative empirical entropy. By information theoretic methods, we carefully derive an estimation of the upper confidence bound for empirical entropy minimization, and leverage it in our minimum entropy upper confidence bound (ME-UCB) algorithm to balance exploration and exploitation. Theoretically, we prove that ME-UCB has a regret upper bound of $O(1)$, which surpasses existing submodular UCB algorithms. Our extensive experiments with both a synthetic and real-world dataset empirically demonstrate that our ME-UCB algorithm outperforms other state-of-the-art approaches.**

I. Introduction

Recently, crowdsourcing has emerged as a cheap yet effective paradigm for soliciting useful information from the public crowd to accomplish a wide spectrum of tasks (e.g., image labelling, sentiment analysis, entity resolution) that were traditionally conducted by the specialized few. A typical crowdsourcing system is operated by an online crowdsourcing platform such as Amazon Mechanical Turk (AMT)[1], which matches workers with tasks, and remunerates them based on their efforts and performances. Nowadays, due to the surging demands for big data, the need for crowdsourcing inevitably increases significantly. Reportedly, around 50,000 new workers join AMT each year[2]. While such expansion brings more popularity and diversity to the platforms, it also calls for more efficient algorithms to properly match the workers and tasks.

Thus far, many prior works [1–13] have devoted efforts in worker recruitment for crowdsourcing. However, today's crowdsourcing systems still face a fundamental unsolved problem of how to select the proper set of workers with *a priori unknown worker reliability*. Usually in practice, worker reliability is affected by a variety of complicated real-world factors, such as expertise, effort level, as well as many others, and is thus difficult to be precisely estimated when the worker joins the platform. Without such reliability information, it is challenging for the platform to select the set of workers that will finish crowdsourcing tasks with satisfactory quality.

To address this problem, we propose to adopt the *empirical entropy* of workers' results as a measurement of the execution quality of crowdsourcing tasks. As described in information theory, the concept of information entropy evaluates the uncertainty of a random variable given the probability distribution of it. Empirical entropy, on the other hand, measures the uncertainty of a set of sampled values of a random variable. In a crowdsourcing system, as long as a fair number of workers return relatively accurate results, minimizing the empirical entropy could help gain more confidence in the final aggregated results, and thus improve the task execution quality. Therefore, in this paper, we take the perspective of the crowdsourcing platform, and aim at designing a *worker selection mechanism that minimizes the empirical entropy* of workers' results.

In practice, a crowdsourcing platform oftentimes has to deal with sequentially arriving tasks that need to be completed timely. Naturally, such sequential arrival combined with tasks' real-time execution requirements makes the platform operate in a round-based manner, in which a set of workers is selected in each round to execute the crowdsourcing tasks that arrive in the same round. Under such setting, the platform will have to effectively balance the *exploration* for new workers and *exploitation* for experienced ones, which is rather challenging to achieve. To tackle this challenge, we formulate the worker selection problem as a *combinatorial multi-armed bandit (C-MAB)* problem, which treats each worker as an arm, and aims to learn the best combination of arms that minimize the cumulative empirical entropy.

Although there already exist a vast body of literatures [14–17] addressing C-MAB problems under various contexts, directly applying them in our problem setting does not necessarily guarantee the best performance. In fact, our objective function of minimizing the empirical entropy is submodular in nature, which thus makes our C-MAB problem belong to the family of submodular bandit problems. Traditional submodular bandit algorithms [18, 19] are well-known to guarantee a $O(\sqrt{T})$ regret upper bound given RKHS kernels of

[*] Corresponding author.

[1] https://www.mturk.com/

[2] https://www.cloudresearch.com/resources/blog/how-many-amazon-mturk-workers-are-there/

the submodular function. However, these algorithms are with high computational complexity, which oftentimes involve a large amount of matrix multiplications, and the RKHS kernel for the empirical entropy function is hard to determine as well.

Therefore, we augment the widely adopted Upper Confidence Bound (UCB) approach to obtain our *minimum entropy UCB (ME-UCB)* algorithm, which selects arms according to the past empirical rewards and a careful estimation of the upper bound of the confidence. Specifically, we fully capture the relationships between the empirical entropy and the actual information entropy by information theoretic methods, and integrate in ME-UCB a tighter estimation of the upper confidence bound for empirical entropy minimization. Our theoretical analysis shows that ME-UCB guarantees $O(1)$ regret bound with a lower computational complexity compared with existing UCB methods.

In summary, our main contributions are listed as follows.

- We introduce empirical entropy as the metric for worker selection in crowdsourcing systems, and formulate the worker selection problem as a combinatorial multi-armed bandit problem that minimizes the cumulative empirical entropy.
- We design an efficient online minimum entropy UCB (ME-UCB) algorithm to address the worker selection problem. Moreover, we theoretically prove that the algorithm has a regret upper bound of $O(1)$, which surpasses any general submodular UCB algorithms.
- We conduct extensive experiments on both a synthetic and a real-world crowdsourcing dataset. Our experiment results show that our ME-UCB algorithm outperforms state-of-the-art baseline algorithms in most cases.

The organization of this paper is as follows. In Section II, we survey state-of-the-art works about C-MAB and worker selection in crowdsourcing. In Section III, we introduce our system model and formulation for the worker selection problem. Then, we introduce a novel ME-UCB algorithm and theoretically evaluate its performance in Section IV. Finally, we carry out experiments both randomly-generated and real-world dataset in Section V, and conclude the paper in Section VI.

II. Related Work

Based on studies about stochastic bandits [20–24], algorithms have been proposed for C-MAB recently. The most common methods for Gaussian or linear C-MABs are UCB-based algorithms [14–17], and they all achieve $O(\log(T))$ regret bounds. Based on the result for C-MAB, more general assumptions have been made when the reward function is submodular. [18] proposed SM-UCB, which applies an RKHS kernel to estimate the regret upper bound according to a Gaussian bandit, and derived an upper bound of $O(\sqrt{T})$. After that, more general assumptions are made about contextual information and volatile arms, and CC-MAB is proposed in that scenario [19]. CC-MAB also achieves $O(\sqrt{T})$ regret bound when Holder continuity holds about the reward function. In this work, we propose a UCB-based algorithm and achieves an $O(1)$ regret upper bound, which is better than the general methods proposed in theoretical works [18, 19].

As the problem of worker selection or recruitment in crowdsourcing systems can be easily formulated as a C-MAB problem, most previous works focusing on that area develop bandit algorithms to solve that problem [1–6]. [1] designs an algorithm for joint worker selection and payment to balance the worker's quality and budget control. [2] applies UCB to estimate the quality of workers and then perform task assignment, and thus provides a regret upper bound of $O(\log T)$. Then, [3] extends worker selection to a more general objective, where the overall quality value can be a nonlinear function of the worker's reliability, while ensuring the α-approximate regret bounded by $O(NLK^3 \ln B)$, where B, N and L are the budget, number of workers and number of options respectively. However, [1–3] all suppose that we can acquire the accurate value of each worker's reliability after each round, which is usually infeasible in practice. In order to address the problem, [4, 5] use majority voting for truth discovery, and introduce a (0,1) loss for workers defined as whether the provided answers are the same as majority. The method well solves the difficulty for defining accurate performance, but can only work when the truth discovery method is majority voting. Moreover, [6] further introduces contextual information, where the workers carry information about their potential performance when entering the platform, and applies exploration-exploitation mechanism to solve the task allocation problem. Different from these previous works, we set the minimization of empirical entropy as objective, which is compatible with any truth discovery methods, because minimizing empirical entropy can be viewed as minimizing uncertainty for the workers' results.

Meanwhile, there are also other approaches aiming at efficient worker selection or task allocation for crowdsourcing [7–13]. The works have different approaches, from developing pricing mechanisms [7], to designing network optimization algorithms [8–11], or applying learning-based techniques [12, 13] to match the workers with tasks with different objectives. However, currently the non-bandit approaches to worker selection have not shown obvious advantage compared to bandit-based algorithms. Rather, C-MAB can better model the worker selection process, as it naturally involves balance between exploration and exploitation of workers. Therefore, we still develop a bandit-based algorithm that can nearly approach the theoretical lower regret bound for entropy-minimization.

III. Preliminaries

A. System Overview

We take into consideration a crowdsourcing system, where a centralized cloud-based platform manages a crowd of W participating workers, denoted as $\mathcal{W} = \{1, 2, \cdots, W\}$. The whole crowdsourcing procedure consists of T rounds, where each round consists of a categorical classification task. The potential answer for the task is drawn from a finite discrete set, denoted as $\mathcal{A}$. We assume that all tasks in the entire task belong to the same domain and the worker answers tasks with stable performance. For example, the platform asks the workers to record the total car flow passing a road intersection in a time slot, and the task repeats for T rounds. The set of possible answers is $\{1, 2, \cdots, M\}$.

In each round, as depicted in Figure 1, the following processes are performed.

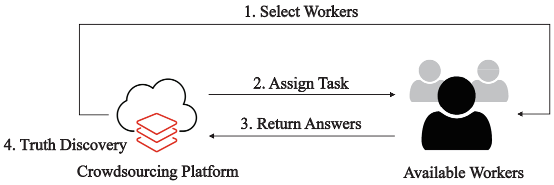


Fig. 1: System model for a single round.

- Due to limited budget, each data collection round t requires us to select a set of N_t workers, denoted as $\mathcal{S}_t$, from the set of workers that are available for the job, denoted as $\mathcal{W}_t$. Clearly, $\mathcal{S}_t$ is a subset of $\mathcal{W}_t$ and $|\mathcal{S}_t| = N_t$.
- Then, the platform assigns assigns the task to the workers $\mathcal{S}_t$ selected at the current round.
- After acquiring the task, each selected worker w finishes their job and reports their answer a_w^t to the platform among a finite set of possible answers $\mathcal{A}$. The answer a_w^t by worker w can be viewed as a random variable drawn from an unknown multinomial distribution Π_w. We let $\mathcal{A}_t$ denote the collection of answers provided by the workers in $\mathcal{S}_t$ at tth round, i.e., $\mathcal{A}_t = \{a_w^t : w \in \mathcal{S}_t\}$.
- Finally, after collecting answers from the selected workers, the platform aggregates their answers and outputs an answer for the task by some truth discovery techniques [25–29].

B. Problem Description

While the method of truth discovery varies, we aim at minimizing the information uncertainty for the answers given by the workers. We can evaluate the empirical entropy according to the answers given by workers at the end of each round, and then by applying the entropy-minimization algorithms discussed in Section IV, we can select the workers so that the overall uncertainty for answers could be minimized. The empirical entropy can be formally defined as follows.

Definition 1. *Provided the answer set $\mathcal{A}_t$ at round t, the empirical entropy of $\mathcal{A}_t$, $\widehat{H}(\mathcal{A}_t)$, is defined as follows*[3].

$$\widehat{H}(\mathcal{A}_t) = -\sum_{k=1}^{p} \frac{\sum_{i\in\mathcal{S}_t} \mathbb{I}\{a_i^t = k\}}{N_t} \log \frac{\sum_{i\in\mathcal{S}_t} \mathbb{I}\{a_i^t = k\}}{N_t}, \quad (1)$$

where $\mathbb{I}\{\cdot\}$ is the indicator function with

$$\mathbb{I}\{x\} = \begin{cases} 1, & x \text{ is true} \\ 0, & \text{otherwise} \end{cases}. \quad (2)$$

Meanwhile, we can also define the entropy generated by a set of workers as follows.

Definition 2. *The entropy for a selected set of workers $\mathcal{S}_t$ is defined as follows.*

$$H(\mathcal{S}_t) = -\sum_{k=1}^{p} \frac{\sum_{i\in\mathcal{S}_t} \mathbb{P}\{a_i^t = k\}}{N_t} \log \frac{\sum_{i\in\mathcal{S}_t} \mathbb{P}\{a_i^t = k\}}{N_t}. \quad (3)$$

For generality, we allow that the available set of workers is variable during the whole crowdsourcing process, but stay fixed during one round of crowdsourcing task. Moreover, it is worth noting that our system only focuses on selecting workers and assigning tasks for a crowdsourcing platform, with the privacy guarantee and worker incentive being independently solved by present and futural works.

Our objective of minimizing the empirical entropy is equivalent to maximizing the negative entropy so that the reward can be positive in each round. From information theory, it is clear that the entropy provided by uniform distribution, or random guess, is larger than any other distributions. Therefore, we let Π_0 denote the uniform distribution, and define the negative entropy as follows.

Definition 3. *The negative entropy $G(\mathcal{S}_t)$ given the selected workers $\mathcal{S}_t$ and the empirical negative entropy $\widehat{G}(\mathcal{A}_t)$ given the answer set $\mathcal{A}_t$ are defined as follows.*

$$\begin{cases} G(\mathcal{S}_t) = H(\Pi_0) - H(\mathcal{S}_t) = \log p - H(\mathcal{S}_t) \\ \widehat{G}(\mathcal{A}_t) = H(\Pi_0) - \widehat{H}(\mathcal{A}_t) = \log p - \widehat{H}(\mathcal{A}_t) \end{cases}. \quad (4)$$

C. Problem Formulation

During the entire data collection process, a sequence of empirical negative entropies $\widehat{G}(\mathcal{A}_1), \widehat{G}(\mathcal{A}_2), \cdots, \widehat{G}(\mathcal{A}_t)$ is attained. Our objective is to maximize the overall expectation of empirical negative entropy by selecting optimal combinations of workers at each round. In short, the problem can be formulated as the following optimization program.

$$\max \sum_{t=1}^{T} \mathbb{E}[\widehat{G}(\mathcal{A}_t)] \quad (5a)$$

$$\text{s.t.} \quad a_i^t \sim \Pi_i, \forall i \in \mathcal{S}_t, t \in \{1, 2, \cdots, T\} \quad (5b)$$

$$\mathcal{S}_t \subseteq \mathcal{W}_t, |\mathcal{S}_t| = N_t, \forall i \in \mathcal{S}_t, t \in \{1, 2, \cdots, T\} \quad (5c)$$

Equation (5a) is the expectation of cumulative empirical entropy. Equation (5b) indicates that the answers are drawn from multinomial distributions, and Equation (5c) shows that the selected set of workers must be among the available workers, with cardinality N_t. Therefore, the problem of worker selection is a sequential-decision problem, where we need to balance the exploration and exploitation of different workers. Thus, we can further formulate the problem as the entropy minimization bandit (EM-MAB) in Definition 4.

Definition 4. *The entropy-minimization bandit (EM-MAB) problem can be formulated as follows.*

- ***Arms**. A worker w among the available set of workers $\mathcal{W}_t$ can be viewed as an **arm**, and in each round t we need to select a set of workers $\mathcal{S}_t$. In our EM-MAB problem, $\mathcal{S}_t$ is called a **super arm**.*
- ***Round reward**. The **round reward** is the negative entropy $\widehat{H}(\mathcal{A}_t)$, which depends on the distributions $\Pi_i, \forall i \in \mathcal{S}_t$. In EM-MAB, we need to maximize the cumulative reward, i.e., the cumulative negative entropy shown in Equation (5a).*

[3] If $\sum_{i\in\mathcal{S}_t} \mathbb{I}\{a_i^t = k\} = 0$, we omit that item, as $\lim_{x\to 0} x \log x = 0$.

IV. Algorithm Design and Analysis

As the problem of worker selection has been formulated as an EM-MAB problem, a solution to EM-MAB can solve the original worker recruitment problem consequently.

A. Greedy Algorithm

C-MAB problems with linear or Lipschitz-continuous rewards can be easily solved by various state-of-the-art algorithms [14–17], either with or without a feedback of reward for single arms. However, in this problem, due to the submodularity of entropies, normal algorithms designed for linear C-MAB problems cannot be directly applied.

An intuitive approach to the C-MAB problem with submodular reward is the greedy algorithm. The main idea of greedy algorithm is to repeatedly select the worker that provides greatest increment to the objective function until the maximum allowable number of workers have been selected. The increment for the negative entropy by each worker can be easily defined by the difference between the negative entropy in the current round and the negative entropy excluding the worker's result. We name the average increment caused by a worker w until round t as the average reward for this worker, denoted as $\bar{\mu}_w(t)$. It is defined as follows.

Definition 5. *The average reward $\bar{\mu}_w(t)$ for worker w at time slot t is*

$$\bar{\mu}_w(t) = \frac{1}{T_w(t)} \sum_{j=1}^{t} \widehat{G}(\mathcal{A}_t) - \widehat{G}(\mathcal{A}_t \setminus \{w\}), \tag{6}$$

where $T_w(t) = \sum_{j=1}^{t} \mathbb{I}(w \in \mathcal{A}_t)$ means the number of times worker w have been selected.

The greedy algorithm is shown in Alg. 1. We use two online variables, $\bar{\mu}_w$ and T_w to record the average reward and exploration times for each worker w, respectively. The average reward $\bar{\mu}_w$ are initialize to be ∞ for each worker w to make sure that each worker would be explored at least once. In each round, we first sort the workers by $\bar{\mu}_w$ in decreasing order (line 3), and then select the top N_t workers as the worker set (line 4). After the selected workers return answers, we calculate the round reward (line 6), and update $\bar{\mu}_w$ and T_w for each selected workers (line 7-9).

Algorithm 1: Greedy Algorithm for Worker Selection

```
1 Initialize $\bar{\mu}_w \leftarrow \infty, T_w \leftarrow 0, \forall w \in \mathcal{W}$;
2 for $t = 1$ to $T$ do
3     Sort $\mathcal{W}_t$ by $\bar{\mu}_w$ decreasingly;
4     Select top $N_t$ workers in $\mathcal{W}_t$ as $\mathcal{S}_t$;
5     Workers in $\mathcal{S}_t$ return answers $\mathcal{A}_t$;
6     Calculate the round reward $\widehat{G}(\mathcal{A}_t)$;
7     foreach $w \in \mathcal{S}_t$ do
8         $\bar{\mu}_w \leftarrow \frac{T_w}{T_w+1}\bar{\mu}_w + \frac{1}{T_w+1}(\widehat{G}(\mathcal{A}_t) - \widehat{G}(\mathcal{A}_t \setminus \{x\}))$;
9         $T_w \leftarrow T_w + 1$;
```

The main drawback of the greedy algorithm is that it can easily fall into a local optimal. If a professional worker is explored and the the first-round reward for the worker is very low, then the worker probably will never be selected again. Consequently, the optimal combination composed of several professional workers will never be achieved, resulting in a local optimal. Therefore, we design the ME-UCB algorithm in the next section that solves the problem.

B. Minimum-Entropy Upper Confidence Bound Algorithm

Upper confidence bound (UCB) algorithm is a solution for avoiding the local optimal of the greedy algorithm. The UCB algorithm introduces an additive positive item for the average reward of worker. Such additional item decreases over time, meaning a confidence bound for the reward of the worker.

Recent researches [18, 19] have provided generalized UCB algorithms for general submodular MABs provided the RKHS kernel of the objective function. However, determining the RKHS kernel for a submodular function is hard, and these algorithms include massive matrix inversions and multiplications, causing tremendous computational complexity when the number of workers and number of possible answers increases.

In order to address the EM-MAB efficiently and optimally, we propose *minimum-entropy upper confidence bound (ME-UCB) algorithm*, as shown in Alg. 2.

Algorithm 2: ME-UCB Algorithm for Worker Selection

```
Initialize $\bar{\mu}_w \leftarrow \infty, T_w \leftarrow 0, \forall w \in \mathcal{W}$;
for $t = 1$ to $T$ do
    foreach $w \in \mathcal{W}_t$ do
        $\widehat{\mu}_w \leftarrow \bar{\mu}_w + \frac{\beta(t)}{\sigma_{\mathcal{N}}^2(t) T_w}$;
    Sort $\mathcal{W}_t$ by $\widehat{\mu}_w$ decreasingly;
    Select top $N_t$ workers in $\mathcal{W}_t$ as $\mathcal{S}_t$;
    Workers in $\mathcal{S}_t$ return answers $\mathcal{A}_t$;
    Calculate the round reward $\widehat{G}(\mathcal{A}_t)$;
    foreach $w \in \mathcal{S}_t$ do
        $\bar{\mu}_w \leftarrow \frac{T_w}{T_w+1}\bar{\mu}_w + \frac{1}{T_w+1}(\widehat{G}(\mathcal{A}_t) - \widehat{G}(\mathcal{A}_t \setminus \{x\}))$;
        $T_w \leftarrow T_w + 1$;
```

The difference between ME-UCB and the greedy algorithm is that we design a particular upper confidence bound for the negative entropy, $\frac{\beta(t)}{\sigma_{\mathcal{N}}^2(t) T_w}$, where $\beta(t)$ is a function of t shared by all workers, and should strictly satisfy

$$\beta(t) \leq t. \tag{7}$$

For specific applications, $\beta(t)$ can be set accordingly to achieve the best performance. A larger $\beta(t)$ indicates more attention on confidence bound when selecting workers,i.e., exploration, and a smaller $\beta(t)$ means more attention on previous behavior, i.e., exploitation. Specifically, $\beta(t) = 0$ makes ME-UCB devolve to the greedy algorithm. $\sigma_{\mathcal{N}}$ is a scalar related to the number of selected workers, defined as:

$$\sigma_{\mathcal{N}}^2(t) = \frac{N_t - 1}{32 \log^2(N_t - 1)}, \tag{8}$$

and T_w is the number of times worker w has been chosen until round t, as introduced before.

C. Performance Analysis

In 1998, U. Feige proved that for any submodular function maximization problem, no polynomial-time algorithm can achieve a better approximation than greedy algorithm [30], and greedy algorithm gives an approximation ratio of $1-1/e$. In order to evaluate the efficiency of our algorithm, we denote $\mathbb{E}[\widehat{G}(\mathcal{S}_j)] = \bar{G}(\mathcal{S}_j)$, and define the regret $R(T)$ for the ME-UCB algorithm as:

$$R(T) = \sum_{j=1}^{T} (1 - 1/e) \cdot \mathrm{OPT}_j - \bar{G}(\mathcal{S}_j). \tag{9}$$

Without loss of generality, we suppose $N_t = N$. OPT_j in Equation (9) is the best expected result provided by a set of workers with cardinality N at round j, i.e.,

$$\mathrm{OPT}_j = \max_{\mathcal{S}^* \in \mathcal{W}_j : |\mathcal{S}^*| = N} \bar{G}(\mathcal{S}^*). \tag{10}$$

As the workers are selected sequentially in one round, we use $\mathcal{S}_{i,j}$ to denote the first i workers that are selected in round j, i.e., the workers with i largest $\widehat{\mu}_w(t)$. Then, let $x_{i+1,j}$ denote the $i+1$th worker to be selected in round j. Based on the work by L. Chen, et al. [18], the regret can be upper bounded by the Lemma 1.

Lemma 1. *The regret w.r.t.* T *is bounded by*

$$R(T) \leq \sum_{j=1}^{T} R_{N,j} = \sum_{j=1}^{T} \sum_{i=1}^{N} r_{i,j}, \tag{11}$$

where

$$r_{i,j} = \sup_a \Delta(a|\mathcal{S}_{i,j}) - \Delta(x_{i+1,j}|\mathcal{S}_{i,j}), \tag{12}$$

with

$$\Delta(a|\mathcal{S}_{i,j}) = \bar{G}(\mathcal{S}_{i,j} \cup \{a\}) - \bar{G}(\mathcal{S}_{i,j}). \tag{13}$$

Proof. See Appendix A for proof. □

As shown in the lemma, the overall regret can be bounded by the cumulative difference of reward for workers between the ME-UCB policy and the optimal greedy policy which has ground truth for the probability distributions of the workers' selections. In order to bound the cumulative difference $\sum_{j=1}^{T} \sum_{i=1}^{N} r_{i,j}$, we first show that the probability for the difference between the real and empirical entropy can be bounded by the value of a Gaussian tail distribution's CDF. Then, we can apply the CDF value to estimate the upper bound for regret using the technique for bounding regrets for Gaussian bandits [31, 32].

Derived from the relationship between two empirical entropies, we have the following theorem that bounds the empirical entropy from real entropy with some probability.

Theorem 1. *For any set of workers* τ*, and for any* $\varepsilon \geq 0$*, the gap between empirical entropy and real entropy satisfies*

$$\mathbb{P}[H(\tau) - \widehat{H}(\tau) \leq -\varepsilon] \leq \exp\left(-\frac{\varepsilon^2 N}{8 \log^2 N}\right),$$

and

$$\mathbb{P}[H(\tau) - \widehat{H}(\tau) \geq \varepsilon] \leq \exp\left(-\frac{\left[\varepsilon - \log \frac{|\mathcal{T}|+N-1}{N}\right]^2 N}{8 \log^2 N}\right).$$

Therefore, combining the above two inequalities can we derive the following bound as

$$\mathbb{P}[|H(\tau) - \widehat{H}(\tau)| \leq \varepsilon] \geq 1 -$$
$$1 \left\{ \exp\left(-\frac{\varepsilon^2 N}{8 \log^2 N}\right) + \exp\left(-\frac{\left[\varepsilon - \log \frac{|\mathcal{T}|+N-1}{N}\right]^2 N}{8 \log^2 N}\right) \right\}.$$

Proof. According to [33], for any set of workers τ, the following relationship between $\widehat{H}(\tau)$ and $H(\tau)$ always holds.

$$\mathbb{E}[\widehat{H}(\tau)] \geq H(\tau),$$

and

$$\lim_{N \to \infty} \mathbb{E}[\widehat{H}(\tau)] = H(\tau).$$

As shown in [34], let H_D be the empirical entropy with N samples, H_S be a subsample of H_D with M samples, with probability at least $1 - \alpha$, we have

$$H_D \geq H_S - \sqrt{\frac{8(N-M) \log \frac{1}{\alpha}}{MN}} \log M,$$

and with probability at least $1 - \alpha$,

$$\begin{aligned} H_D \leq & H_S + \sqrt{\frac{8(N-M) \log \frac{1}{\alpha}}{MN}} \log M \\ & + \log\left(1 + \frac{(c-1)(N-M)}{M(N-1)}\right). \end{aligned}$$

where c is the number of choices. By taking $N \to \infty$ and applying $H_D \to H$, $H_S = \widehat{H}$, the theorem holds. □

According to Theorem 1, we have known the gap between empirical entropy and real entropy. Then, we can estimate the negative entropy gain for a worker after several times of exploration in Lemma 2. We use $\widehat{G}(\tau)$ to denote the single-step empirical entropy including a worker a, and $\widehat{G}'(\tau)$ to denote the single-step entropy discarding the worker a. Similarly, $\bar{G}(\tau)$ means the average entropy including a worker a, and $\bar{G}'(\tau)$ means the average entropy discarding a. Let $\mu_a = \widehat{G}(\tau) - \widehat{G}'(\tau)$, $\tilde{\mu}_a = \bar{G}(\tau) - \bar{G}'(\tau)$, and $\bar{\mu}_{a,s}$ be the average value of μ_a after s times of sampling.

Lemma 2. *Multi-step error bound for a single worker* a *is*

$$\mathbb{P}(\mu_a - \bar{\mu}_{a,s} \geq \varepsilon) \leq \exp\left\{-\frac{(\mu_{\mathcal{N}} + \varepsilon \sigma_{\mathcal{N}}^2 s)^2}{2\sigma_{\mathcal{N}}^2}\right\}, \tag{14}$$

where

$$\mu_{\mathcal{N}} = \log \frac{|\tau| + N - 1}{N}, \sigma_{\mathcal{N}}^2 = \frac{N-1}{32 \log^2 (N-1)}, \tag{15}$$

and N *is sufficiently large.*

Proof. Single step error between empirical entropy gain and average entropy gain is bounded by

$$\begin{aligned}&\mathbb{P}(\mu_a - \tilde{\mu}_a \geq \varepsilon)\\ =&\mathbb{P}([\widehat{G}(\tau) - \widehat{G}'(\tau)] - [\bar{G}(\tau) - \bar{G}'(\tau)] \geq \varepsilon)\\ =&\mathbb{P}([\widehat{G}(\tau) - \bar{G}(\tau)] + [\bar{G}'(\tau) - \widehat{G}'(\tau)] \geq \varepsilon)\\ \leq&\mathbb{P}\left[\widehat{G}(\tau) - \bar{G}(\tau) \geq \delta\right] \mathbb{P}\left[\bar{G}'(\tau) - \widehat{G}'(\tau) \geq \varepsilon - \delta\right],\end{aligned}$$

As $f(n) = \frac{n}{\log^2(n)}$ is monotonic-increasing when n is large ($n \geq 7$ when the base is 2), and according to Theorem 1,

$$\begin{aligned}&\mathbb{P}\left[\widehat{G}(\tau) - \bar{G}(\tau) \geq \delta\right] \mathbb{P}\left[\bar{G}'(\tau) - \widehat{G}'(\tau) \geq \varepsilon - \delta\right]\\ \leq&\exp\left\{-\frac{N-1}{8\log^2(N-1)}\left(\left[\delta - \log\frac{|\mathcal{T}|+N-1}{N}\right]^2 + [\varepsilon-\delta]^2\right)\right\}\\ \leq&\exp\left\{-\frac{N-1}{16\log^2(N-1)}\left(\varepsilon - \log\frac{|\mathcal{T}|+N-1}{N}\right)^2\right\},\end{aligned}$$

when $\delta = \frac{1}{2}\left(\varepsilon + \log\frac{|\mathcal{T}|+N-1}{N}\right)$ and N is sufficiently large. Thus, the tail of $\mu_a - \widehat{\mu}_a$ is upper bounded by a $\mathcal{N}(\mu_\mathcal{N}, \sigma_\mathcal{N}^2)$ normal distribution.

Therefore, multi-step error is bounded by

$$\begin{aligned}\mathbb{P}(\mu_a - \bar{\mu}_{a,s} \geq \varepsilon) &\leq \mathbb{E}\left(e^{\lambda(\mu_a - \bar{\mu}_{a,s})}\right)e^{-\lambda\varepsilon}\\ &\leq \int_{-\infty}^{\infty}\exp\left\{-\frac{(x-\mu_\mathcal{N})^2 + 2\lambda x/n}{2\sigma_\mathcal{N}^2} - \lambda\varepsilon\right\}\mathrm{d}x\\ &= \exp\left\{\frac{\lambda^2 - 2\lambda\mu_\mathcal{N}s}{2\sigma_\mathcal{N}^2 s^2} - \lambda\varepsilon\right\}\\ &= \exp\left\{-\frac{(\mu_\mathcal{N} + \varepsilon\sigma_\mathcal{N}^2 s)^2}{2\sigma_\mathcal{N}^2}\right\},\end{aligned}$$

where the large equality is because $\lambda = -\mu_\mathcal{N}s - \varepsilon\sigma_\mathcal{N}^2 s^2$. □

Theorem 2. *The regret of ME-UCB is upper bounded by*

$$R(T) \leq |\mathcal{W}|N\left[\frac{1}{2\varepsilon} + 2\sqrt{2}\cdot\frac{\varepsilon - \Delta_{i,k}^{\max}}{\sigma_\mathcal{N}}\right] \tag{16}$$

for any $0 < \varepsilon < 1$.

Proof. See Appendix B for proof. □

Therefore, the regret $R(T)$ for ME-UCB can be bounded by a constant that is irrelevant to T. Compared to previous results, which are usually $O(\sqrt{T})$, our algorithm achieves better theoretical performance on maximizing the cumulative negative entropy than general algorithms for submodular rewards.

V. Experiments

We conduct experiments on both simulation environment and a large-scale crowdsourcing dataset to validate our ME-UCB algorithm.

A. Evaluation Methodology

a) Dataset: We use the RTE-6 dataset from National Institute of Standards and Technology (NIST), U.S. [35]. The RTE-6 dataset consists of human-labeled answers for binary-classification textual entailment recognition tasks, and is made publicly available. For each worker w, their probability of answer selection is $\Pi_w = (p_0, p_1)$. We use p_0 to denote the probability of choosing the right answer, and p_1 to denote the probability of choosing the wrong answer. For each textual entailment task, we use the majority voting to select the ground truth, and calculate p_0 and p_1 for each worker.

Moreover, we also use randomly-generated data to compare the performance of different algorithms.

b) Baseline Algorithm: We select several popular and frequently-used general algorithms as our baselines. We include both algorithms for Gaussian combinatorial bandits and an algorithm for combinatorial bandits with general rewards. The algorithms are briefly introduced as follows.

- **Greedy algorithm**. The greedy algorithm has been introduced in section IV-A. Due to its simplicity, it has been widely used in general cases for sequential decisive problems with submodular rewards.
- **Gaussian UCB**. Gaussian UCB is an upper-confidence bound-based algorithm designed for both linear bandits or combinatorial bandits. The expected reward with confidence bound for each sub-arm (for combinatorial bandits) i at round t is defined as follows.

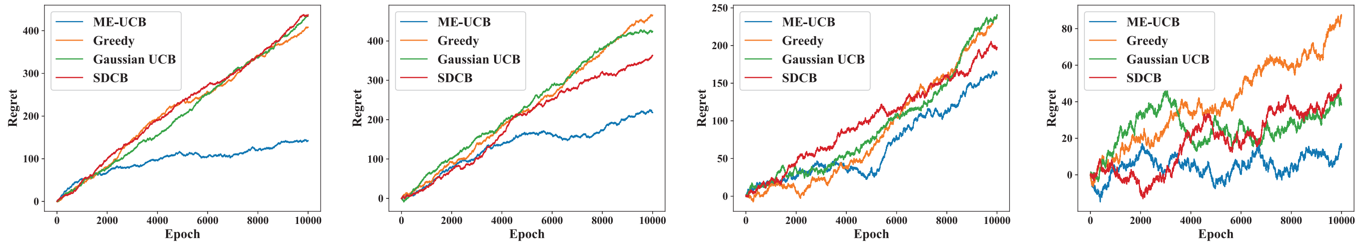
  $$\bar{\mu}_i(t) = \widehat{\mu}_i(t-1) + \sqrt{\frac{2}{T_i(t-1)}\log\frac{1}{\delta}}, \tag{17}$$

  where δ is a monotonic-decreasing function of t. In the test case, we define $\frac{1}{\delta} = 1 + t\log^2(t)$, which is value that is widely adapted and achieves good result.
- **SDCB** [36]. Stochastically dominant confidence bound (SDCB) is a lower-confidence bound-based algorithm designed for combinatorial bandits with general reward functions. After exploration for each worker for at least one time, at the t-th round, for each arm i, it defines a distribution $\bar{D}_i$ with CDF $\bar{F}_i(x)$ as follows.

  $$\bar{F}_i(x) = \begin{cases}\max\left\{\widehat{F}_i(x) - \sqrt{\frac{3\ln t}{2T_i}}, 0\right\}, & 0 \leq x < 1,\\ 1, & x = 1,\end{cases} \tag{18}$$

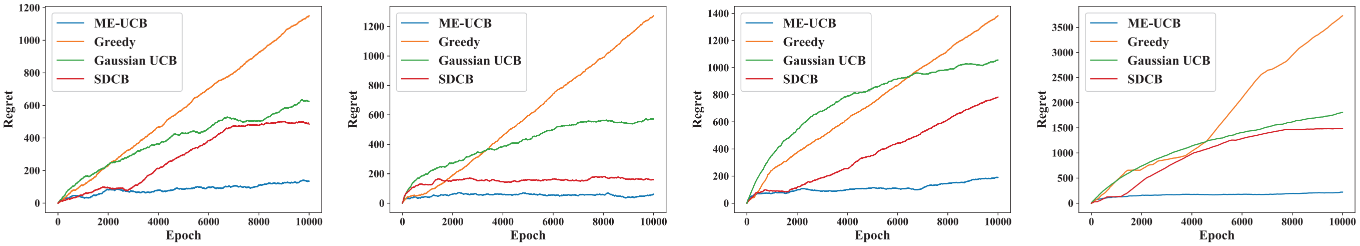
  where $\widehat{F}_i(x)$ means the fraction of observed outcomes from arm i that are no larger than x. At t-th round, with the distributions for each arms, an oracle is adapted to select a super arm S_t, i.e., $S_t = \mathrm{Oracle}(\bar{D})$, where $\bar{D} = \bar{D}_1 \times \bar{D}_2 \times \cdots \bar{D}_N$. In the test case, we use a greedy oracle to select the arms from distributions, i.e., $\mathrm{Oracle}(\bar{D}) = \arg\max_{S_t}\sum_{i\in S_t}\mathbb{E}(\bar{D}_i)$.

c) Empirical Regret: Although theoretical analysis show that no algorithm achieves better than $O(T)$ approximation ratio, the result only holds in the worst case. If we still define regret as Equation (9), the regret would always be negative in



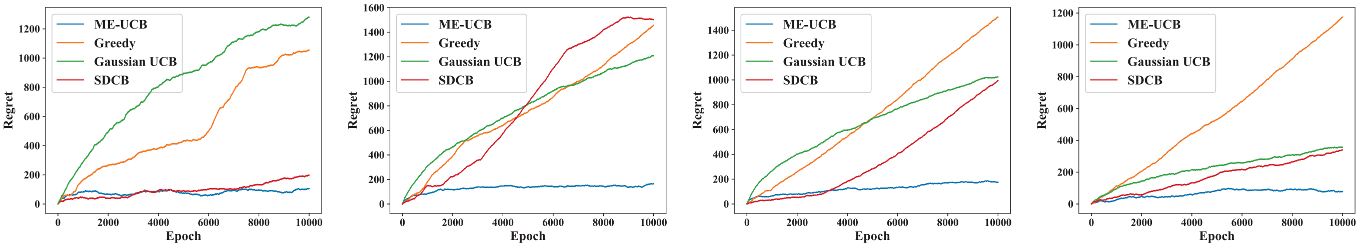
(a) $P = 0.3$ (b) $P = 0.5$ (c) $P = 0.7$ (d) $P = 0.9$

Fig. 2: Regret (R) w.r.t. Epoch (T) with average correct probabilities ($P = \mathbb{P}[a_i = 1]$), when number of workers is 20, number of choices is 10, and in each round 5 workers are selected.



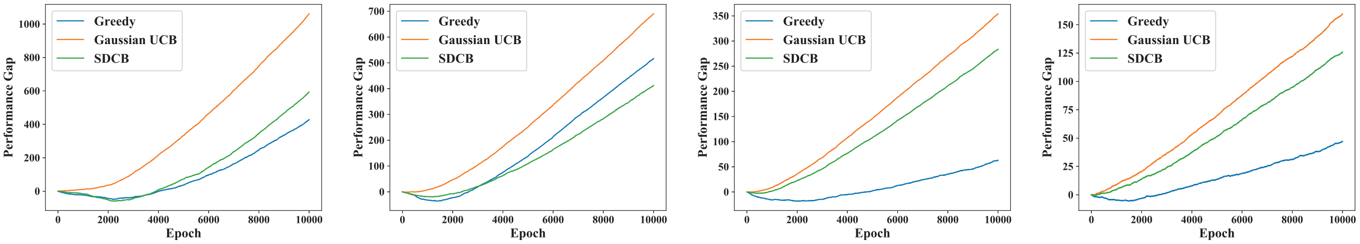
(a) 10 workers (b) 15 workers (c) 20 workers (d) 25 workers

Fig. 3: Regret (R) w.r.t. Epoch (T) with randomly-generated probabilities ($P = \mathbb{P}[a_i = k]$), when number of workers varies, number of choices is 10, and in each round 5 workers are selected.



(a) 3 selected workers (b) 6 selected workers (c) 9 selected workers (d) 15 selected workers

Fig. 4: Regret (R) w.r.t. Epoch (T) with randomly-generated probabilities ($P = \mathbb{P}[a_i = k]$), when number of selected workers varies, number of workers is 20, and number of choices is 10.



(a) 20 selected workers (b) 40 selected workers (c) 80 selected workers (d) 120 selected workers

Fig. 5: Performance gap between baseline algorithms and ME-UCB on RTE dataset when the number of selected workers varies.

most cases. Therefore, throughout our experiments, we define the empirical regret $R(T)$ just by the traditional regret, as

$$R(T) = \sum_{j=1}^{T} \mathrm{OPT}_j - \bar{G}(\mathcal{A}_j). \tag{19}$$

Using the new empirical definition for regret, we can perform the following evaluation for our algorithm.

B. Evaluation Result

Referring the usual UCB values for other UCB algorithms, we set $\beta(t) = \frac{1}{10}\sqrt{\log(t)}$. We first use randomly-generated data to validate the performance of our ME-UCB algorithm. In different scenarios, the average quality for the answers of the workers may be different. Therefore, we generate several sets of probability distributions for workers to represent the difference in average qualities. Without loss of generality, we let option 1 for each task be the ground truth, and suppose that $\mathbb{P}[a_i = 1] \geq 1/p$, where p is the number of options. The assumption means that the workers all have some knowledge about the field, and thus always provide answers that are better than random guess, where $\mathbb{P}[a_i = k] = 1/p$. We vary the value of $\mathbb{P}[a_i = 1]$ and randomly generate the values for $\mathbb{P}[a_i = k], k \neq 1$, and observe the performance gap between different algorithms. Figure 2 shows the performance for the four algorithms when the average correct probability, denoted by P, varies from 0.3 to 0.9, when the number of workers is fixed as 20, number of choices is 10, and in each round 5 workers are selected. Results show that Gaussian UCB algorithm and greedy algorithm usually perform the worst, and our ME-UCB algorithm usually outperforms other algorithms in different cases. Specifically, when the correct probabilities for workers are low, the performance gap between ME-UCB and other combinatorial bandit algorithms is large. In contrast, when the correct probabilities become higher, the regret for a wrong choice of worker becomes lower, and thus the performance of different algorithms varies insignificantly.

Then we keep the average value for $\mathbb{P}[a_i = 1]$ to be around 0.5, but randomly-generated, the number of choices to be 10, and the number of selected workers in each round to be 5, and vary the number of total workers. As shown in Figure 3, the performances of greedy algorithm, Gaussian UCB and SDCB are all unstable, but the performance of ME-UCB is very stable when the number of total workers varies and the situation changes. The empirical regret is about $O(\log T)$ by observation and is rather stable across T.

Finally, we stick to 20 total workers, 10 choices and randomly generated $\mathbb{P}[a_i = 1]$ with average value 0.5, and vary the number of selected workers. Figure 4 shows with different number of selected workers, ME-UCB still outperforms other 3 baseline algorithms. SDCB performs well when the number of selected workers is low, but is outperformed by Gaussian UCB when we need to select more workers.

For the RTE dataset, the task is a binary classification task and includes 164 workers. In this situation, finding the optimal combination of workers is impossible, and we cannot use regret to evaluate the performance of different algorithms. Instead, we use the cumulative reward given by ME-UCB algorithm, and compare the performance gap between 3 baseline algorithms and ME-UCB. The performance gap is defined by $\sum_{j=1}^{T} \bar{G}(\mathcal{A}_j^{\mathrm{ME-UCB}}) - \bar{G}(\mathcal{A}_j)$. Therefore, the larger the performance gap is, the worse the performance of the algorithm is. If the performance gap is negative, it indicates that the performance for the current algorithm is better than ME-UCB, and vice versa. If we observe Figure 5, we can come to the conclusion that in the long term, ME-UCB is guaranteed to outperform all 3 baseline algorithms in cumulative reward defined by negative entropy. However, in the short term, SDCB and greedy algorithm can outperform ME-UCB, but with relatively low difference.

VI. Conclusion

In this paper, we study the worker selection problem in a crowdsourcing system for minimizing cumulative empirical entropy. We formulate the problem as an EM-MAB problem and provide a general greedy algorithm. Then we develop an ME-UCB algorithm for worker selection, and prove that the regret is upper bounded by $|\mathcal{W}|N\left[\frac{1}{2\varepsilon} + 2\sqrt{2}\cdot\frac{\varepsilon-\Delta_{i,k}^{max}}{\sigma_{\mathcal{N}}}\right]$ for any $0 < \varepsilon < 1$ except for the theoretical lower bound of $1/e$ reward loss. Finally, we conduct experiments on randomly-generated datasets and real-life RTE dataset to validate the performance of our algorithm compared to several baseline algorithms. Both theoretical analysis and experiments show the advantage of our algorithm than other state-of-the-art algorithms focusing on submodular MABs.

Acknowledgement

This work was supported in part by National Key R&D Program of China 2018AAA0101200, in part by NSFC China (No. 61902244, U20A20181, 61960206002, 61822206, 62020106005, 61829201, 62041205, 61532012), in part by Shanghai Municipal Science and Technology Commission Grant 19YF1424600.

Appendix A
Proof of Lemma 1

Proof. As $\mathcal{A}_j = \{x_{1,j}, x_{2,j}, \cdots, x_{N,j}\}$ is the set of workers selected sequentially at round j, we can use $\mathcal{A}_{i,j} = \{x_{1,j}, \cdots, x_{i,j}\}$ to denote the first i workers that are selected at round j. Moreover, let $\mathcal{A}^*$ denote the set of optimal workers such that for any $\mathcal{A} \subseteq \mathcal{W}, |\mathcal{A}| = N$,

$$\bar{G}(\mathcal{A}^*) \geq \bar{G}(\mathcal{A}).$$

Therefore, according to submodularity of entropy, we have

$$\begin{aligned}\bar{G}(\mathcal{A}^*) \leq \bar{G}(\mathcal{A}^* \cup \mathcal{A}_{i,j}) \leq \bar{G}(\mathcal{A}_{i,j}) + \sum_{a\in\mathcal{A}^*} \Delta(a|\mathcal{A}_{i,j})\\ \leq \bar{G}(\mathcal{A}_{i,j}) + N\sup_a\{\Delta(a|\mathcal{A}_{i,j})\}\\ = \bar{G}(\mathcal{A}_{i,j}) + N[r_{i+1,j} + \Delta(x_{i+1,j}|A+i,j)],\end{aligned}$$

where

$$\begin{aligned}r_{i+1,j} = \sup_a \Delta(a|\mathcal{A}_{i,j}) - \Delta(x_{i+1,j}|\mathcal{A}_{i,j})\\ =\bar{G}(\mathcal{A}_{i,j}) + N\left[R_{i+1,j} - R_{i,j} + \bar{G}(\mathcal{A}_{i+1,j}) - \bar{G}(\mathcal{A}_{i,j})\right].\end{aligned}$$

Here $R_{i,j}$ is the cumulative for $r_{i,j}$ at the jth round, defined as follows:

$$R_{i,j} = \sum_{\alpha=1}^{i} r_{\alpha,j}.$$

Therefore, we can put $r_{i+1,j}$ into the expression for $\bar{G}(\mathcal{A}^*)$ and get that

$$\begin{aligned}\bar{G}(\mathcal{A}^*) - \bar{G}(\mathcal{A}_{i,j}) \leq N[R_{i+1,j} - R_{i,j} + \bar{G}(\mathcal{A}_{i+1,j}) - \bar{G}(\mathcal{A}_{i,j})]\\ \leq N\{R_{i+1,j} - R_{i,j} - [\bar{G}(\mathcal{A}^*) - \bar{G}(\mathcal{A}_{i+1,j})] + [\bar{G}(\mathcal{A}^*) - \bar{G}(\mathcal{A}_{i,j})]\}.\end{aligned}$$

Let $\delta_{i,j} = \bar{G}(\mathcal{A}^*) - \bar{G}(\mathcal{A}_{i,j})$ to simplify expression, and therefore

$$\delta_{i,j} \leq N[R_{i+1,j} - R_{i,j} - \delta_{i+1,j} + \delta_{i,j}],$$

and thus,

$$\delta_{i,j} \leq R_{i,j} - R_{i-1,j} + \left(1 - \frac{1}{N}\right)\delta_{i-1,j}.$$

As $\bar{G}(\emptyset) = 0$, we know that $\delta_{0,j} = \bar{G}(\mathcal{A}^*)$ for any j. Write the expression for $\delta_{i,j}$ iteratively, and we can get

$$\begin{aligned}
&\delta_{i,j} \leq R_{i,j} - R_{i-1,j} + \left(1 - \frac{1}{N}\right) \delta_{i-1,j} \\
\leq& R_{i,j} - R_{i-1,j} + \left(1 - \frac{1}{N}\right) \left[R_{i-1,j} - R_{i-2,j} + \left(1 - \frac{1}{N}\right) \delta_{i-2,j}\right] \\
\leq& \sum_{\alpha=0}^{i-1} \left(1 - \frac{1}{N}\right)^{\alpha} (R_{i-k,j} - R_{i-k-1,j}) + \left(1 - \frac{1}{N}\right)^{i} \delta_{0,j} \\
=& R_{i,j} - \frac{1}{N} \sum_{\alpha=0}^{i-1} \left(1 - \frac{1}{N}\right)^{\alpha} R_{i-1-k,j} + \left(1 - \frac{1}{N}\right)^{i} \delta_{0,j} \\
\leq& R_{i,j} + e^{-\frac{i}{N}} \bar{G}(\mathcal{A}^*).
\end{aligned}$$

Let $i = N$, then $\mathcal{A}_{i,j} = \mathcal{A}_{N,j} = \mathcal{A}_j$ and

$$\bar{G}(\mathcal{A}^*) - \bar{G}(\mathcal{A}_j) \leq R_{N,j} + e^{-1}\bar{G}(\mathcal{A}^*),$$

which indicates that

$$\left(1 - \frac{1}{e}\right) \bar{G}(\mathcal{A}^*) - \bar{G}(\mathcal{A}_j) \leq R_{N,j} = \sum_{i=1}^{N} r_{i,j}.$$

Therefore, we come to the conclusion that

$$R(T) \leq \sum_{j=1}^{T} R_{N,j} = \sum_{j=1}^{T} \sum_{i=1}^{N} r_{i,j}.$$

And we complete the proof. □

Appendix B
Proof of Theorem 2

Proof. As $r_{i,j} = \Delta(a_{i,j}|\mathcal{A}_{i,j}) - \Delta(x_{i+1,j}|\mathcal{A}_{i,j}) = \bar{G}(\{a_{i,j}\} \cup \mathcal{A}_{i,j}) - \bar{G}(\{x_{i+1,j}\} \cup \mathcal{A}_{i,j})$, we have

$$\begin{aligned}
&R(T) = \sum_{j=1}^{T} \sum_{i=1}^{N} r_{i,j} \\
=& \sum_{k=1}^{N_w} \sum_{j=1}^{T} \sum_{i=1}^{N} \mathbb{P}(x_{i+1,j} = k)[\bar{G}(\{a_{i,j}\} \cup \mathcal{A}_{i,j}) - \bar{G}(\{k\} \cup \mathcal{A}_{i,j})] \\
=& \sum_{k=1}^{N_w} \sum_{j=1}^{T} \sum_{i=1}^{N} \mathbb{E}\left\{\mathbb{I}[x_{i+1,j} = k]\right\} \Delta_{i,j,k},
\end{aligned}$$

where we use $\Delta_{i,j,k}$ to denote $\bar{G}(\{a_{i,j}\} \cup \mathcal{A}_{i,j}) - \bar{G}(\{k\} \cup \mathcal{A}_{i,j})$ for convenience. Then we have

$$\begin{aligned}
&R(T) \\
\leq& \sum_{k=1}^{N_w} \sum_{i=1}^{N} \sum_{j=1}^{T} \Delta_{i,j,k} \mathbb{E}\{\mathbb{I}[\bar{\mu}_{a_{i,j}}(j) + \frac{\beta_j}{\sigma_{\mathcal{N}}^2 s} \leq \mu_{a_{i,j}} - \varepsilon] \\
&+ \mathbb{I}[\bar{\mu}_{x_{i+1,j}} + \frac{\beta_j}{\sigma_{\mathcal{N}}^2 s} \geq \mu_{a_{i,j}} - \varepsilon, x_{i+1,j} = k]\} \\
\leq& \sum_{k=1}^{N_w} \sum_{i=1}^{N} \Delta_{i,k}^{\max} \sum_{j=1}^{T} \mathbb{E}\{\mathbb{I}[\bar{\mu}_{a_{i,j}}(j) + \frac{\beta_j}{\sigma_{\mathcal{N}}^2 s} \leq \mu_{a_{i,j}} - \varepsilon] \\
&+ \mathbb{I}[\bar{\mu}_{x_{i+1,j}} + \frac{\beta_j}{\sigma_{\mathcal{N}}^2 s} \geq \mu_{a_{i,j}} - \varepsilon, x_{i+1,j} = k]\}
\end{aligned}$$

where we use $\Delta_{i,k}^{\max}$ to denote $\max_j \Delta_{i,j,k}$. Then we analyze two parts $\mathbb{E}\left\{\mathbb{I}[\bar{\mu}_{a_{i,j}}(j) + \frac{\beta_j}{\sigma_{\mathcal{N}}^2 s} \leq \mu_{a_{i,j}} - \varepsilon]\right\}$ and $\mathbb{E}\left\{\mathbb{I}[\bar{\mu}_{x_{i+1,j}} + \frac{\beta_j}{\sigma_{\mathcal{N}}^2 s} \geq \mu_{a_{i,j}} - \varepsilon, x_{i+1,j} = k]\right\}$ separately, where ϵ is any real number ranging from 0 to 1. Firstly we bound $\mathbb{E}\left\{\mathbb{I}[\bar{\mu}_{a_{i,j}}(j) + \frac{\beta_j}{\sigma_{\mathcal{N}}^2 s} \leq \mu_{a_{i,j}} - \varepsilon]\right\}$ as follows.

$$\begin{aligned}
&\sum_{j=1}^{T} \mathbb{E}\left\{\mathbb{I}[\bar{\mu}_{a_{i,j}}(j) + \frac{\beta_j}{\sigma_{\mathcal{N}}^2 s} \leq \mu_{a_{i,j}} - \varepsilon]\right\} \\
=& \sum_{j=1}^{T} \mathbb{P}\left\{\bar{\mu}_{a_{i,j}}(j) + \frac{\beta_j}{\sigma_{\mathcal{N}}^2 s} \leq \mu_{a_{i,j}} - \varepsilon\right\} \\
\leq& \sum_{j=1}^{T} \sum_{s=1}^{j} \mathbb{P}\left\{\bar{\mu}_{a_{i,j},s} + \frac{\beta_j}{\sigma_{\mathcal{N}}^2 s} \leq \mu_{a_{i,j}} - \varepsilon\right\} \\
\leq& \sum_{j=1}^{T} \sum_{s=1}^{j} \exp\left\{-\frac{[\mu_{\mathcal{N}} + \varepsilon\sigma_{\mathcal{N}}^2 s + \beta_j]^2}{2\sigma_{\mathcal{N}}^2}\right\} \\
=& \sum_{j=1}^{T} \exp\left\{-\frac{(\mu_{\mathcal{N}} + \beta_j)^2}{2\sigma_{\mathcal{N}}^2}\right\} \sum_{s=1}^{j} \exp\left\{-\frac{\varepsilon^2\sigma_{\mathcal{N}}^4 s^2 + 2\varepsilon\sigma_{\mathcal{N}}^2 s}{2\sigma_{\mathcal{N}}^2}\right\} \\
\leq& \sum_{j=1}^{T} \exp\left\{-\frac{(\mu_{\mathcal{N}} + \beta_j)^2}{2\sigma_{\mathcal{N}}^2}\right\} \int_{0}^{\infty} \exp\left\{-\frac{\varepsilon^2\sigma_{\mathcal{N}}^2 s^2}{2}\right\} \mathrm{d}s \\
\leq& \sum_{j=1}^{T} \exp\left\{-\frac{(\mu_{\mathcal{N}} + \beta_j)^2}{2\sigma_{\mathcal{N}}^2}\right\} \cdot \frac{1}{\sqrt{2}\varepsilon\sigma_{\mathcal{N}}} \\
\leq& \frac{1}{\sqrt{2}\varepsilon\sigma_{\mathcal{N}}} \int_{0}^{\infty} \exp\left\{-\frac{\beta_j^2}{2\sigma_{\mathcal{N}}^2}\right\} \mathrm{d}j \leq \frac{1}{2\varepsilon}.
\end{aligned}$$

And then we bound the right part as follows.

$$\begin{aligned}
&\sum_{j=1}^{T} \mathbb{E}\left\{\mathbb{I}[\bar{\mu}_{x_{i+1,j}} + \frac{\beta_j}{\sigma_{\mathcal{N}}^2 s} \geq \mu_{a_{i,j}} - \varepsilon, x_{i+1,j} = k]\right\} \\
\leq& \sum_{s=1}^{T} \mathbb{E}\left\{\mathbb{I}[\bar{\mu}_{x_{i+1,s}} + \frac{\beta_T}{\sigma_{\mathcal{N}}^2 s} \geq \mu_{a_{i,s}} - \varepsilon]\right\} \\
\leq& \sum_{s=1}^{T} \mathbb{E}\left\{\mathbb{I}[\bar{\mu}_{x_{i+1,s}} - \mu_{x_{i+1,s}} + \frac{\beta_T}{\sigma_{\mathcal{N}}^2 s} \geq \Delta_{i,s} - \varepsilon]\right\} \\
\leq& \sum_{s=1}^{T} \mathbb{P}\left\{|\mu_{x_{i+1,s}} - \bar{\mu}_{x_{i+1,s}}| \geq |\varepsilon + \frac{\beta_T}{\sigma_{\mathcal{N}}^2 s} - \Delta_{i,k}^{max}|\right\} \\
\leq& \sum_{s=1}^{T} 2\exp\left\{-\frac{[\mu_{\mathcal{N}} + |\varepsilon\sigma_{\mathcal{N}}^2 s + \beta_T - \Delta_{i,k}^{max}\sigma_{\mathcal{N}}^2 s|]^2}{2\sigma_{\mathcal{N}}^2}\right\} \\
\leq& 2\sum_{s=1}^{T} \exp\left\{-\frac{[\varepsilon\sigma_{\mathcal{N}}^2 s + \beta_T - \Delta_{i,k}^{max}\sigma_{\mathcal{N}}^2 s]^2}{2\sigma_{\mathcal{N}}^2}\right\} \\
=& 2\sum_{s=1}^{T} \exp\left\{-\frac{[s + \frac{\beta_T}{(\varepsilon - \Delta_{i,k}^{max})\sigma_{\mathcal{N}}^2}]^2}{2\frac{(\varepsilon - \Delta_{i,k}^{max})^2}{\sigma_{\mathcal{N}}^2}}\right\} \\
\leq& 2\int_{-\infty}^{\infty} \exp\left\{-\frac{[s + \frac{\beta_T}{(\varepsilon - \Delta_{i,k}^{max})\sigma_{\mathcal{N}}^2}]^2}{2\frac{(\varepsilon - \Delta_{i,k}^{max})^2}{\sigma_{\mathcal{N}}^2}}\right\} \mathrm{d}s \\
=& 2\sqrt{2} \cdot \frac{\varepsilon - \Delta_{i,k}^{max}}{\sigma_{\mathcal{N}}},
\end{aligned}$$

which leads to the conclusion. □

References

[1] A. Biswas, S. Jain, D. Mandal, and Y. Narahari, “A truthful budget feasible multi-armed bandit mechanism for crowdsourcing time critical tasks,” in *AAMAS*, 2015.

[2] A. Rangi and M. Franceschetti, “Multi-armed bandit algorithms for crowdsourcing systems with online estimation of workers’ ability,” in *AAMAS*, 2018.

[3] G. Gao, J. Wu, M. Xiao, and G. Chen, “Combinatorial multi-armed bandit based unknown worker recruitment in heterogeneous crowdsensing,” in *INFOCOM*, 2020.

[4] H. Zhang, Y. Ma, and M. Sugiyama, “Bandit-based task assignment for heterogeneous crowdsourcing,” *Neural Computation*, vol. 27, no. 11, pp. 2447–2475, 2015.

[5] Y. Liu and M. Liu, “An online learning approach to improving the quality of crowd-sourcing,” *IEEE/ACM Transactions on Networking*, vol. 25, no. 4, pp. 2166–2179, 2017.

[6] S. Klos née Müller, C. Tekin, M. van der Schaar, and A. Klein, “Context-aware hierarchical online learning for performance maximization in mobile crowdsourcing,” *IEEE/ACM Transactions on Networking*, vol. 26, no. 3, pp. 1334–1347, 2018.

[7] W. Liu, Y. Yang, E. Wang, and J. Wu, “Dynamic user recruitment with truthful pricing for mobile crowdsensing,” in *INFOCOM*, 2020.

[8] J.-X. Liu and K. Xu, “Budget-aware online task assignment in spatial crowdsourcing,” *WWW*, 2020.

[9] S. Yang, K. Han, Z. Zheng, S. Tang, and F. Wu, “Towards personalized task matching in mobile crowdsensing via fine-grained user profiling,” in *INFOCOM*, 2018.

[10] X. Wang, R. Jia, X. Tian, and X. Gan, “Dynamic task assignment in crowdsensing with location awareness and location diversity,” in *INFOCOM*, 2018.

[11] Y. Chen, B. Li, and Q. Zhang, “Incentivizing crowdsourcing systems with network effects,” in *INFOCOM*, 2016.

[12] C. H. Liu, Z. Dai, H. Yang, and J. Tang, “Multi-task-oriented vehicular crowdsensing: A deep learning approach,” in *INFOCOM*, 2020.

[13] L. Chen and J. Xu, “Task replication for vehicular cloud: Contextual combinatorial bandit with delayed feedback,” in *INFOCOM*, 2019.

[14] W. Chen, Y. Wang, and Y. Yuan, “Combinatorial multi-armed bandit: General framework and applications,” in *ICML*, 2013.

[15] R. Combes, M. S. Talebi, A. Proutière, and M. Lelarge, “Combinatorial bandits revisited,” in *NIPS*, 2015.

[16] W. Chen, W. Hu, F. Li, J. Li, Y. Liu, and P. Lu, “Combinatorial multi-armed bandit with general reward functions,” in *NIPS*, 2016.

[17] S. Wang and W. Chen, “Thompson sampling for combinatorial semi-bandits,” in *ICML*, 2018.

[18] L. Chen, A. Krause, and A. Karbasi, “Interactive submodular bandit,” in *NIPS*, 2017.

[19] L. Chen, J. Xu, and Z. Lu, “Contextual combinatorial multi-armed bandits with volatile arms and submodular reward,” in *NIPS*, 2018.

[20] W. Ren, J. Liu, and N. B. Shroff, “Exploring k out of top ρ fraction of arms in stochastic bandits,” in *Proceedings of Machine Learning Research*, vol. 89, pp. 2820–2828, 2019.

[21] S. Cayci, A. Eryilmaz, and R. Srikant, “Budget-constrained bandits over general cost and reward distributions,” in *International Conference on Artificial Intelligence and Statistics*, vol. 108, pp. 4388–4398, 2020.

[22] J. Zuo, X. Zhang, and C. Joe-Wong, “Observe before play: Multi-armed bandit with pre-observations,” *SIGMETRICS*, 2019.

[23] S. Wang and L. Huang, “Multi-armed bandits with compensation,” in *NIPS*, 2018.

[24] A. Badanidiyuru, R. Kleinberg, and A. Slivkins, “Bandits with knapsacks,” *J. ACM*, vol. 65, no. 3, 2018.

[25] D. Zhou, Q. Liu, J. C. Platt, C. Meek, and N. B. Shah, “Regularized minimax conditional entropy for crowdsourcing,” *CoRR*, vol. abs/1503.07240, 2015.

[26] C. Miao, W. Jiang, L. Su, Y. Li, S. Guo, Z. Qin, H. Xiao, J. Gao, and K. Ren, “Privacy-preserving truth discovery in crowd sensing systems,” *ACM Trans. Sens. Networks*, vol. 15, no. 1, pp. 9:1–9:32, 2019.

[27] X. Tang, C. Wang, X. Yuan, and Q. Wang, “Non-interactive privacy-preserving truth discovery in crowd sensing applications,” in *INFOCOM*, 2018.

[28] H. Jin, L. Su, and K. Nahrstedt, “Theseus: Incentivizing truth discovery in mobile crowd sensing systems,” in *MobiHoc*, 2017.

[29] H. Jin, L. Su, and K. Nahrstedt, “Centurion: Incentivizing multi-requester mobile crowd sensing,” in *INFOCOM*, 2017.

[30] U. Feige, “A threshold of ln n for approximating set cover,” *J. ACM*, vol. 45, no. 4, p. 634–652, 1998.

[31] N. Srinivas, A. Krause, S. M. Kakade, and M. W. Seeger, “Gaussian process optimization in the bandit setting: No regret and experimental design,” in *ICML*, 2010.

[32] A. Krause and C. S. Ong, “Contextual gaussian process bandit optimization,” in *NIPS*, 2011.

[33] S. Verdú, “Empirical estimation of information measures: A literature guide,” *Entropy*, vol. 21, no. 8, p. 720, 2019.

[34] C. Wang and B. Ding, “Fast approximation of empirical entropy via subsampling,” in *SIGKDD*, 2019.

[35] NIST, “Past rte data.” https://tac.nist.gov/data/RTE/index.html, 2017.

[36] W. Chen, W. Hu, F. Li, J. Li, Y. Liu, and P. Lu, “Combinatorial multi-armed bandit with general reward functions,” in *NIPS*, 2016.