Optimal PAC Multiple Arm Identi cation with Applications to Crowdsourcing

Yuan Zhou
Carnegie Mellon University
yuanzhou@cs.cmu.edu

Xi Chen
UC Berkeley
xichen1987@berkeley.edu

Jian Li
Tsinghua University
lijian83@mail.tsinghua.edu.cn

April 24, 2014

Abstract

We study the problem of selecting K arms with the highest expected rewards in a stochastic N-armed bandit game. Instead of using existing evaluation metrics (e.g., misidentification probability (Bubeck et al., 2013) or the metric in Explore-K(Kalyanakrishnan & Stone, 2010)), we propose to use *the aggregate regret*, which is defined as the gap between the average reward of the optimal solution and that of our solution. Besides being a natural metric by itself, we argue that in many applications, such as our motivating example from the crowdsourcing, the aggregate regret bound is more suitable. We propose a new PAC algorithm, which, with probability at least $1-\delta$, identifies a set of K arms with regret at most

and hence expensive to acquire. Therefore, it is desirable to select the best K workers with the minimum number of queries. This problem can be cast into our top-K arm identification problem, where each worker corresponds to an arm and the mean μ_i characterizes the ith worker's underlying reliability/quality.

More formally, assume that the arms are ordered by their means: $\mu_1 > \mu_2 > \cdots > \mu_n$ and let T be the set of selected arms with size $|T| = K$. We define the *aggregate regret* (or *regret* for short) of T as:

$$\mathcal{L}_T = \frac{1}{K}\left(\sum_{i=1}^{K} \mu_i - \sum_{i \in T} \mu_i\right). \tag{1}$$

Our goal is to design an algorithm with low sample complexity and PAC (Probably Approximately Correct) style bounds. More specifically, given any fixed positive constants ε, δ, the algorithm should be able to identify a set T of K arms with $\mathcal{L}_T \leq \varepsilon$ (we call such a solution an ε-optimal solution), with probability at least $1 - \delta$.

We first note that our problem strictly generalizes the previous work by (Even-Dar et al., 2006; Mannor & Tsitsiklis, 2004) for $K = 1$ to arbitrary positive integer K and hence is referred to as multiple arm identification problem. Although the problem of choosing multiple arms has been studied in some existing work, e.g., (Bubeck et al., 2013; Audibert et al., 2013; Kalyanakrishnan & Stone, 2010; Kalyanakrishnan et al., 2012), our notion of aggregate regret is inherently different from previously studied evaluation metrics such as misidentification probability (MISID-PROB) (Bubeck et al., 2013) and EXPLORE-K (Kalyanakrishnan & Stone, 2010; Kalyanakrishnan et al., 2012). In particular, MISID-PROB controls the probability that the output set T is not exactly the same as the top-K arms; and EXPLORE-K requires to return a set T where, with high confidence, the mean of each arm in T is ε-close to the K-th best arm. As we will explain in the related work section, our evaluation metric is a more suitable objective for many real applications, especially for the aforementioned crowdsourcing application.

We summarize our results in this paper as follows:

1. Section 3 & 4: We develop a new PAC algorithm with sample complexity $O\left(\frac{n}{\varepsilon^2}\left(1 + \frac{\ln(1/\delta)}{K}\right)\right)$ for any positive constants ε, δ, and any $1 \leq K \leq n/2$. For $n/2 \leq K < n$, the sample complexity becomes $O\left(\frac{n-K}{K} \cdot \frac{n}{\varepsilon^2}\left(\frac{n-K}{K} + \frac{\ln 1/\delta}{K}\right)\right)$: The analysis of the algorithm is presented in Section 4. It is interesting to compare this bound with the optimal $O(\frac{n}{\varepsilon^2} \ln \frac{1}{\delta})$ bound for $K = 1$ in (Even-Dar et al., 2006; Mannor & Tsitsiklis, 2004). For $K = 1$ (i.e., selecting the best arm), our result matches theirs. Interestingly, when K is larger, our algorithm suggests that even less samples are needed. Intuitively, a larger K leads to a less stringent constraint for an ε-optimal solution and thus can tolerate more mistakes. Let us consider the following toy example. Assume all the arms have the same mean $1/2$, except for a random one with mean $1/2 + \varepsilon$. If $K = 1$, to obtain an ε-optimal solution, we essentially need to identify the special arm and thus need a lot of samples. However, if K is large, any subset of K arms would work fine since the regret is at most $2\varepsilon/K$. Our algorithm bears some similarity with previous work, such as the halving technique in (Even-Dar et al., 2006; Kalyanakrishnan & Stone, 2010; Karnin et al., 2013) and idea of accept-reject in (Bubeck et al., 2013). However, the analysis is more involved than the case for $K = 1$ and needs to be done more carefully in order to achieve the above sample complexity.

2. Section 5: To complement the upper bound, we further establish a matching lower bound for Bernoulli bandits: for $1 \leq K \leq n/2$, any (deterministic or randomized) algorithm requires at least $\Omega\left(\frac{n}{\varepsilon^2}\left(1 + \frac{\ln(1/\delta)}{K}\right)\right)$ samples to obtain an ε-optimal solution with the confidence $1 - \delta$; for $n/2 \leq K < n$, the lower bound becomes $\Omega\left(\frac{n-K}{K} \cdot \frac{n}{\varepsilon^2}\left(\frac{n-K}{K} + \frac{\ln 1/\delta}{K}\right)\right)$: This shows that our algorithm achieves the optimal sample complexity for Bernoulli bandits and for all values of ε, δ and K. To this end, we show two different lower bounds for $1 \leq K \leq n/2$: $\Omega(\frac{n}{\varepsilon^2})$ and $\Omega(\frac{n}{\varepsilon^2}\frac{\ln(1/\delta)}{K})$. The first bound is established via an interesting reduction from our problem to the basic problem of distinguishing two similar Bernoulli arms (with means $1/2$ and $1/2+\varepsilon$ respectively). The second one can be shown via a generalization of the argument

in (Mannor & Tsitsiklis, 2004) for $K = 1$. The lower bound for $n=2 \leq K < n$ can be easily derived by a reduction to the case for $1 \leq K \leq n=2$.

3. Section 6: Finally, we conduct extensive experiments with both simulated and real data sets. The experimental results demonstrate that, using the same number of samples, our algorithm not only achieves lower regrets but also higher precisions than existing methods . Morever, using our algorithm, the maximum number of samples taken from any individual arm is much smaller than that in the SAR algorithm (Bubeck et al., 2013). This property is particularly desirable for crowdsourcing applications since it can be quite problematic, at least time-consuming, to test a single worker with too samples.

2 Related Works

Multi-armed bandit problems have been extensively studied in the machine learning community over the past decade (see for example (Auer et al., 2002a,b; Beygelzimer et al., 2011; Bubeck & Cesa-Bianchi, 2012) and the references therein). In recent years, the multiple arm identification problem has received much attention and has been investigated under different setups. For example, the work (Even-Dar et al., 2006; Mannor & Tsitsiklis, 2004; Audibert et al., 2010; Karnin et al., 2013) studied the the special case when $K = 1$. When $K > 1$, Bubeck et al.(2013) proposed a SAR (Successive Accepts and Rejects) algorithm which minimizes the *misidentification probability*, (i.e., $\Pr(T \neq \{1, \ldots, K\})$, denoted as MISID-PROB), given a fixed budget (of queries). Another line of research (Kalyanakrishnan et al., 2012; Kalyanakrishnan & Stone, 2010) proposed to select a subset T of arms, such that with high probability, for all arms $i \in T$, $_i > _K - $, where $_K$ is the mean of the K-th best arm. We refer this metric to as the EXPLORE-K metric.

Our notion of aggregate regret is inherently different from MISID-PROB and EXPLORE-K, and is a more suitable objective for many real applications. For example, MISID-PROB requires to identify the exact top-K arms, which is more stringent. When the gap of any consecutive pair $_i$ and $_{i+1}$ among the first $2K$ arms is extremely small (e.g., $o(\frac{1}{n})$), it requires a huge amount (e.g., $!(n^2)$) of samples to make the misidentification probability less than (Bubeck et al., 2013). While in our metric, any K arms among the first $2K$ arms consititute an -optimal solution. In crowdsourcing applications, our main goal is not to select the exact top-K workers, but a pool of good enough workers with a small number of samples. We note that the *expected simple regret*, $\frac{1}{K}(\sum_{i=1}^{K} {}_i - \mathbf{E}[\sum_{i \in T} {}_i])$, was also considered in a number of prior works (Audibert et al., 2010; Bubeck et al., 2013; Audibert et al., 2013). In (Audibert et al., 2010; Bubeck et al., 2013), the expected simple regret was shown to be sandwiched by $\Delta \cdot$MISID-PROB and MISID-PROB (for $K = 1$), where $\Delta = {}_1 - {}_2$. However, Δ can be arbitrarily small, hence MISID-PROB can be an arbitrarily bad bound for the simple regret. It is worthwhile noting that it is possible to obtain an expected simple regret of with at most $O(n^{2=})$ samples, using the semi-bandit regret bound in (Audibert et al., 2013)[1]. In constrast, the goal of this paper is to develop an efficient algorithm to achieve an -regret with high probability, which is a stronger requirement than obtaining an -expected simple regret.

To compare our aggregate regret with the EXPLORE-K metric, let us consider another example where ${}_1, \ldots, {}_{K-1}$ are much larger than ${}_K$ and ${}_{K+i} > {}_K - $ for $i = 1, \ldots, K$. It is easy to see that the set $T = \{K+1, \ldots, 2K\}$ also satisfies the requirement of EXPLORE-K [2]. However, the set T is far away from the optimal set with the aggregate regret much larger than . In crowdsourcing, the labeling performance can downgrade to a significant extent if the best set of workers (e.g., ${}_1, \ldots, {}_{K-1}$ in the example) is left out of the solution.

[1] The result in (Audibert et al., 2013) was stated in terms of expected accumulative regret (i.e., the expected regret over Z time slots). By setting the number of time slots Z to be $\frac{n}{K^2}$, and choosing a random action as the final solution among Z actions (see e.g., (Bubeck et al., 2009)), one can get an expected simple regret of ϵ.

[2] For this particular instance, it is unlikely that the algorithms proposed in (Kalyanakrishnan et al., 2012; Kalyanakrishnan & Stone, 2010) would choose $\{K+1, \ldots, 2K\}$ as the solution, even though it is a valid solution under their EXPLORE-K metric. However, it is not clear, from their theoretical analysis, how good their solution is, collectively, as compared with the best K arms.

Algorithm 1 Optimal Multiple Arm Identification (OptMAI)

1: Input: $n; K; Q$.
2: Initialization: Active set of arms $S_0 = \{1; : : : ; n\}$; $= e^{0:2} \cdot \frac{3}{4}$; set of top arms $T_0 = \emptyset$. Let $r = 0$
3: while $|T_r| < K$ and $|S_r| > 0$ do
4: if $|S_r| \geq 4K$ then
5: $S_{r+1} = QE(S_r; K; \ ^r(1 - \ )Q)$
6: $T_{r+1} = \emptyset$
7: else
8: $(S_{r+1}; T_{r+1}) = AR(S_r; T_r; K; \ ^r(1 - \ )Q)$
9: end if
10: $r = r + 1$.
11: end while
12: Output: The set of the selected K-arms T_r.

Algorithm 2 Quartile-Elimination(QE) $(S; K; Q)$

1: Input: $S; K; Q$.
2: Sample each arm $i \in S$ for $Q_0 = \frac{Q}{|S|}$ times and let b_i be the empirical mean of the i-th arm.
3: Find the first quartile (lower quartile) of the empirical mean b_a, denoted by b.
4: Output: The set $V = S \backslash \{i \in S : b_i < b\}$.

Algorithm 3 Accept-Reject(AR) $(S; T; K; Q)$

1: Input: $S; T; K; Q$ and $s = |S|$.
2: Sample each arm $i \in S$ for $Q_0 = \frac{Q}{|S|}$ times and let b_i be the empirical mean of the i-th arm.
3: Let $K' = K - |T|$. Let $b_{(K')}$ and $b_{(K'+1)}$ be the K'-th and $(K'+1)$-th largest empirical means, respectively. Define the empirical gap for each arm $i \in S$:

$$\Delta_i = \max(b_i - b_{(K'+1)}; b_{(K')} - b_i) \tag{2}$$

4: while $|T| < K$ and $|S| > 3s/4$ do
5: Let $a \in \arg\max_{i \in S} \Delta_i$ and set $S = S \backslash \{a\}$.
6: if $b_a \geq b_{(K'+1)}$ then
7: Set $T = T \cup \{a\}$.
8: end if
9: end while
10: Output: The set S and T.

3 Algorithm

In this section, we describe our algorithm for the multiple arm identification problem. Our algorithm OptMAI (Algorithm 1) takes three positive integers $n; K; Q$ as the input, where n is the total number of arms, K is the number of arms we want to choose and Q is an upper bound on the total number of samples [3]. In Section 4, we show that $Q = O \quad \frac{n}{2} \quad 1 + \frac{\ln(1/ \ )}{K}$ suffices to obtain an -optimal solution with probability at least $1 - \ $. OptMAI consists of two stages, the *Quartile-Elimination (QE) stage* (line 4-6) and the *Accept-Reject (AR) stage* (line 8).

The QE stage proceeds in rounds. Each QE round calls the QE subroutine in Algorithm 2, which requires three parameters $S; K$ and Q. Here, S is the set of arms which we still want to pull and Q is the total number

[3] If Algorithm 1 stops at round $r = R$, the total number of samples is $(1 - \beta^R)Q < Q$.

of samples required in this round. We sample each arm in S for $Q{=}|S|$ times and then discard *a quarter* of arms with the minimum empirical mean [4]. We note that in each call of the QE subroutine, we pass different Q values (exponentially decreasing). This is critical for keeping the total number of samples linear and achieving the optimal sample complexity. See Algorithm 1 for the setting of the parameters. The QE stage repeatedly calls the QE subroutine until the number of remaining arms is at most $4K$.

Now, we enter the AR stage, which also runs in rounds. Each AR round (Algorithm 3) requires four parameters, $S; T; K; Q$, where $S; K; Q$ have the same meanings as in QE and T is the set of arms that we have decided to include in our final solution and thus will not be sampled any more. In each AR subroutine (Algorithm 3), we again sample each arm for $Q{=}|S|$ times. We define the *empirical gap* for the i-th arm to be the absolute difference between the empirical mean of the i-th arm and the K'-th (or $K'+1$-th) largest empirical mean, where $K' = K - |T|$ (see Eq.(2)). We remove a quarter of arms with the largest empirical gaps. There are two types of those removed arms: those with the largest empirical means, which are included in our final solution set T, and those with the smallest empirical means, which are discarded from further consideration.

Remark 3.1. *We would like to mention that the naive uniform sampling algorithm, which takes the same number of samples from each arm and chooses the K arms with the largest empirical means, does not achieve the optimal sample complexity. In general, it requires at least $\Omega(n \log n)$ samples, which is $\log n$ factor worse than our optimal bound. See Appendix E for a detailed discussion.*

Remark 3.2. *To achieve the desired asymptotic bound on regret and sample complexity, the AR stage can be substituted by a simpler process which takes a uniform number of samples from each arm and chooses the K arms with the largest empirical means. The details can be found in Appendix D. We choose to present the AR subroutine in this section because 1) it also meets the theoretical bound in Section 4; 2) the AR stage shows a much better empirical performance.*

4 Bounding the Regret and the Sample Complexity

We analyze the regret achieved by our algorithm. Firstly, let us introduce some necessary notations. For any positive integer C, we use $[C]$ to denote the set $\{1; 2; \dots; C\}$. For any subset S of arms, let $\mathrm{ind}_i(S)$ be the arm in S with the i-th largest mean. We use $\mathrm{val}_C(S)$ to denote the average mean of the C best arms in S, i.e., $\mathrm{val}_C(S) \triangleq \frac{1}{C}\sum_{i=1}^{C} \mathrm{ind}_i(S)$. Let $\mathrm{tot}_C(S) = C \cdot \mathrm{val}_C(S)$ be the total sum of the means of the C best arms in S. We first consider one QE round. Suppose S is the set of input arms and V is the output set. We first show that the average mean of the K best arms in V is at most worse that that in S, for some appropriate (depending on Q and $|S|$).

Lemma 4.1. *Assume that $K \le |S|{=}4$ and let V be the output of* QE$(S; K; Q)$ *(Algorithm 2). For every > 0, with probability $1 -$, we have that $\mathrm{val}_K(V) \ge \mathrm{val}_K(S) -$, where $=$ $\frac{|S|}{Q}\ 10 + \frac{4\ln(2=\ )}{K}$:*

The basic idea of the proof goes as follows. Let $p =$ $_{\mathrm{ind}_{|S|=2}(S)}$ be the median of the in S and $= \min_{i\in V}(b_i)$ be the minimum *empirical mean* for the selected arms in V. For each arm i among the top K arms in S, we define the random variable $X_i = 1\{b_{\mathrm{ind}_i(S)} < p + \frac{}{2}\}$, where $1\{\cdot\}$ is the indicator function (i.e., $1\{\text{true}\} = 1$ and $1\{\text{false}\} = 0$). Let $X = \frac{1}{K}\sum_{i=1}^{K}(\ \mathrm{ind}_i(S) - p)X_i$. We further define two events $\mathcal{E}_1 = \{X \le \ \}$ and $\mathcal{E}_2 = \{\ < p + \frac{}{2}\}$. Intuitively, $\mathcal{E}_2$ says that the threshold , as the third quartile of the empirical means, is not much larger than the expected median value (i.e., p). When $\mathcal{E}_2$ happens, $\mathcal{E}_1$ would give an upper bound on the regret $\mathrm{val}_K(S) - \mathrm{val}_K(V)$. We formalize this idea by first showing that the events $\mathcal{E}_1$ and $\mathcal{E}_2$ together imply that $\mathrm{val}_K(V) \ge \mathrm{val}_K(S) -$. Then we prove that with the value defined in the lemma statement, Each of $\mathcal{E}_1$ and $\mathcal{E}_2$ holds with probability at least $1 - \frac{}{2}$. By a simple union bound, our proof is completed. The details are presented in the appendix.

[4] The empirical mean of an arm is the average reward of the arm, over all samples in this round.

Secondly, we provide the regret bound for the AR algorithm in the following lemma with the proof presented in the appendix.

Lemma 4.2. *Let* $(S';T')$ *be the output of the algorithm* $\mathrm{AR}(S;T;K;Q)$. *For every* $\epsilon > 0$, *with probability* $1-\delta$, *we have that*

$$\mathrm{tot}_{K-|T'|}(S') + \mathrm{tot}_{|T'|}(T') \geq \mathrm{tot}_{K-|T|}(S) + \mathrm{tot}_{|T|}(T) - \epsilon K,$$

where $\delta = \sqrt{\frac{|S|}{Q}\left(4 + \frac{\log(2/\delta)}{K}\right)}$.

In each round of the AR-stage with $S = S_r$ and top arms $T = T_r$, the value $\frac{\mathrm{tot}_{K-|T|}(S)+\mathrm{tot}_{|T|}(T)}{K}$ is the best possible average mean of K arms. Lemma 4.2 provides an upper bound for the gap between this value on the output $(T';S')$ by AR and the best possible one. Applying this bound over all rounds would further imply that this value of the output of Algorithm 1 is not far away from that of the real top-K arms. With Lemma 4.1 and Lemma 4.2 in place, we prove the performance of Algorithm 1 in the next theorem.

Theorem 4.3. *For every* $\epsilon > 0$, *with probability at least* $1-\sqrt{\delta}$, *the output of OptMAI algorithm* T *is an* ϵ*-optimal solution (i.e.,* $\mathrm{val}_K(T) \geq \mathrm{val}_K([n]) - \epsilon$*) with* $Q = O\left(\frac{n}{\epsilon^2}\left(1 + \frac{\ln 1/\delta}{K}\right)\right)$.

Theorem 4.3 also provides us the sample complexity of Algorithm 1 for any pre-fixed positive values ϵ and δ, as stated in the next corollary.

Corollary 4.4. *For any positive constants* $\epsilon, \delta > 0$, *it suffices to run Algorithm 1 with*

$$Q = O\left(\frac{n}{\epsilon^2}\left(1 + \frac{\ln(1/\delta)}{K}\right)\right). \tag{3}$$

in order to obtain an ϵ*-optimal solution with probability* $1-\delta$. *In other words, the sample complexity of the algorithm is bounded by* $O\left(\frac{n}{\epsilon^2}\left(1 + \frac{\ln(1/\delta)}{K}\right)\right)$ *from above.*

For $n/2 \leq K < n$, we can easily obtain a better sample complexity as follows.

Theorem 4.5. *For any* $\epsilon > 0$ *and* $n/2 \leq K < n$, *with probability at least* $1-\delta$, *there is an algorithm that can find an* ϵ*-optimal solution* T *(i.e.,* $\mathrm{val}_K(T) \geq \mathrm{val}_K([n]) - \epsilon$*) and the number of samples used is at most at most*

$$O\left(\frac{n-K}{K} \cdot \frac{n}{\epsilon^2}\left(\frac{n-K}{K} + \frac{\ln 1/\delta}{K}\right)\right).$$

Proof. Instead of directly finding the best K arms, we attempt to find the worst $n-K$ arms. First, we can see that for any $\epsilon', \delta' > 0$, we can find a set T' of $n-K$ arms such that

$$\left(\sum_{i\in T'} \mu_i - \sum_{i=K+1}^{n} \mu_i\right) \leq (n-K)\,\epsilon'; \tag{4}$$

(we call such a set T' an ϵ'*-worst solution*) with probability $1-\delta'$, using at most $O\left(\frac{n}{\epsilon'^2}\left(1 + \frac{\ln(1/\delta')}{n-K}\right)\right)$ samples. This can be done by constructing a new multiple arm identification instance and run OPTMAI on the new instance. In the new instance, there is an arm with mean $1-\mu_i$ if the original instance consists of an arm with mean μ_i. Sampling from this arm can be simulated by sampling from the corresponding original arm (if we get a sample of value x from the original arm, we use $1-x$ as the sample for the new arm). It is easy to see that for any ϵ-optimal solution T' for the new instance, we have that $\sum_{i=K+1}^{n}(1-\mu_i) - \sum_{i\in T'}(1-\mu_i) \leq (n-K)\,\epsilon'$ (by definition), which is equivalent to an ϵ'-worst solution for the original instance.

By setting $\epsilon' = \frac{K}{n-K}\cdot\epsilon$ and $T = [n] \setminus T'$, we can see that(4) implies that $\sum_{i=1}^{K}\mu_i - \sum_{i\in T}\mu_i \leq K\epsilon$. The theorem follows. □

5 A Matching Lower Bound

In this section, we provide lower bounds for Bernoulli bandits where the reward of the i-th arm follows a Bernoulli distribution with mean $_i$. We prove that for any underlying $\{\ _i\}_{i=1}^n$ and any randomized algorithm $\mathcal{A}$, the expected number of samples Q required to identify an -optimal solution with probability $1-$ is at least max $\Omega\ \frac{n\ln(1=\ )}{^2K}\ ;\Omega\ \frac{n}{2}\ = \Omega\ \frac{n}{2}\ \frac{\ln(1=\ )}{K}+1$: According to Corollary 4.4, for Bernoulli bandits, our algorithm achieves this lower bound of the sample complexity. In particular, we separate the proof into two parts: in the first part, we show that $Q \geq \Omega\ \frac{n}{2}$; and $Q \geq \Omega\ \frac{n\ln(1=\ )}{^2K}$ in the second.

5.1 First Lower Bound: $Q \geq \Omega\ \frac{n}{\epsilon^2}$

Theorem 5.1. *Fix a real number , integers K; n; where 0 < < 0:01 and 10 ≤ K ≤ n=2. Let $\mathcal{A}$ be a possibly randomized algorithm, so that for any set of n Bernoulli arms with means $_1;\ _2;:::;\ _n$,*

- *$\mathcal{A}$ takes at most Q samples in expectation;*
- *with probability at least 0:8, $\mathcal{A}$ outputs a set T of size K with $\mathrm{val}_K(T) \geq \mathrm{val}_K([n]) -$.*

Then, we have that $Q \geq \Omega(\frac{n}{2})$.

The high level idea of the proof of Theorem 5.1 is as follows. Suppose there is an algorithm $\mathcal{A}$ which can find an -optimal solution with probability at least 0:8 and uses at most Q samples in expectation. We show that we can use $\mathcal{A}$ as a subroutine to construct an algorithm $\mathcal{B}$, which can distinguish whether a single Bernoulli arm has mean 1=2 or $1=2+4$ with at most $\frac{200Q}{n}$ samples (Lemma 5.2). We utilize the well known fact that, for any algorithm (including $\mathcal{B}$), distinguishing such a Bernoulli arm requires at least $\Omega(\frac{1}{2})$ samples. Hence, we must have that $\frac{200Q}{n} \geq \Omega(\frac{1}{2})$, which gives the desired lower bound for Q.

Formally, we show in the following lemma how to construct $\mathcal{B}$, using $\mathcal{A}$ as a subroutine.

Lemma 5.2. *Let $\mathcal{A}$ be an algorithm in Theorem 5.1. There is an algorithm $\mathcal{B}$, which correctly outputs whether a Bernoulli arm X has the mean $\frac{1}{2}+4$ or the mean $\frac{1}{2}$ with probability at least 0:51, and $\mathcal{B}$ makes at most $\frac{200Q}{n}$ samples.*

Assuming the existence of an algorithm $\mathcal{A}$ stated in Theorem 5.1, we construct the algorithm $\mathcal{B}$ as follows. Keep in mind that the goal of $\mathcal{B}$ is to distinguish whether a given Bernoulli arm (denoted as X) has mean 1=2 or $1=2+4$.

Algorithm 4 Algorithm $\mathcal{B}$ (which calls $\mathcal{A}$ as a subroutine)

1: Choose a random subset $S \subseteq [n]$ such that $|S| = K$ and then choose a random element $j \in S$.
2: Create n artificial arms as follows: For each $i \in [n]; i \neq j$, let $_i = \frac{1}{2}+4$ if $i \in S$, let $_i = \frac{1}{2}$ otherwise.
3: Simulate $\mathcal{A}$ as follows: whenever $\mathcal{A}$ samples the i-th arm:
   (1) If $i = j$, we sample the Bernoulli arm X (recall X is the arm which $\mathcal{B}$ attempts to separate);
   (2) Otherwise, we sample the arm with mean $_i$.
4: If the arm X is sampled by less than $\frac{200Q}{n}$ times and $\mathcal{A}$ returns a set T such that $j \notin T$, we decide that X has the mean of $\frac{1}{2}$; otherwise we decide that X has the mean of $\frac{1}{2}+4$.

We note that the number of samples of $\mathcal{B}$ increases by one whenever X is sampled. Since $\mathcal{B}$ stops and outputs the mean $\frac{1}{2}+4$ if the number of samples on X reaches $\frac{200Q}{n}$, $\mathcal{B}$ takes at most $\frac{200Q}{n}$ samples from X. The intuition why the above algorithm can separate X is as follows. If X has mean $1=2+4$, X is no different from any other arm in S. Similarly, if X has mean 1=2, X is the same as any other arm in $[n] \setminus S$. If $\mathcal{A}$ satisfies the requirement in Theorem 5.1, $\mathcal{A}$ can identify a significant proportion of arms with mean $1=2+4$. So if X has mean $1=2+4$, there is a good chance (noticeably larger than 0:5) that X will be chosen by $\mathcal{A}$. In the appendix, we formally prove the correctness of $\mathcal{B}$, i.e., it can correctly output the mean of X with probability at least 0.51; and thus conclude the proof of Lemma 5.2.

The second step of the proof of Theorem 5.1 is a well-known lower bound on the expected sample complexity for separating a single Bernoulli arm (Chernoff, 1972; Anthony & Bartlett, 1999).

Lemma 5.3. *Fix such that* $0 < < 0{:}01$ *and let* X *be a Bernoulli random variable with mean being either* $\frac{1}{2}+4$ *or* $\frac{1}{2}$*. If an algorithm* $\mathcal{B}$ *can output the correct mean of* X *with probability at least* $0{:}51$*, then expected number of samples performed by* $\mathcal{B}$ *is at least* $\Omega(\frac{1}{2})$*.*

By combining Lemma 5.2 and Lemma 5.3, we have $\frac{200Q}{n} \geq \Omega(\frac{1}{2})$; and therefore prove the claim that $Q \geq \Omega(\frac{n}{2})$ in Theorem 5.1.

5.2 Second Lower Bound: $Q \geq \Omega\ \frac{n\ln(1/\delta)}{\epsilon^2 K}$

Lemma 5.4. *Fix real numbers ; such that* $0 < ; \leq 0{:}01$*, and integers* $K; n$ *such that* $K \leq n{=}2$*. Let* $\mathcal{A}$ *be a deterministic algorithm (i.e., the only randomness comes from the arms), so that for any set of* n *Bernoulli arms with means* $_1; _2; \ldots; _n$*,*

- $\mathcal{A}$ *makes at most* Q *samples in expectation;*
- *with probability at least* $1 - $ *,* $\mathcal{A}$ *outputs a set* T *of size* K *with* $\mathrm{val}_K(T) \geq \mathrm{val}_K([n]) - $ *.*

Then, we have that $Q \geq \frac{n\ln(1{=})}{20000\ ^2 K}$*.*

Now, we provide a sketch of our proof, which generalizes the previous proof for the lower bound when $K = 1$ (Mannor & Tsitsiklis, 2004). Let $t = \lfloor \frac{n}{K} \rfloor \geq 2$ and we divide the first tK arms into t groups, where the j-th group consists of the arms with the indices in $[(j-1)K+1; jK]$. We first construct t hypotheses $H_1; H_2; \ldots; H_t$ as follows. In H_1, we let $_i = 1{=}2 + 4$ for arms in the first group and let $_i = 1{=}2$ for the remaining arms. In H_j, where $2 \leq j \leq t$, we let $_i = 1{=}2 + 4$ when i is in the first group, $_i = 1{=}2 + 8$ when i is in the j-th group, and $_i = 1{=}2$ otherwise.

For each $j \in [t]$, let $\Pr_{H_j}[\cdot]$ ($\mathsf{E}_{H_j}[\cdot]$ resp.) denote the probability of the event in $[\cdot]$ (the expected value of the random variable in $[\cdot]$ resp.) under the hypothesis H_j. Let $\tilde{q}_j$ be the total number of samples taken from the arms in the j-th group. By an averaging argument, there must exist a group $j_0 \geq 2$ such that $\mathsf{E}_{H_1}[\tilde{q}_{j_0}] \leq \frac{Q}{t-1} \leq \frac{2Q}{t}$.

After fixing j_0, we focus on the hypothesis H_1 and H_{j_0}. Let $\mathrm{val}_K^{H_j}(T)$ ($\mathrm{val}_K^{H_j}([n])$ resp.) be the $\mathrm{val}_K(T)$ value ($\mathrm{val}_K([n])$ resp.) computed using values defined in hypothesis H_j. Note that $\mathrm{val}_K^{H_j}(T)$ (for any j) is always well defined no matter which hypothesis is the true underlying probability measure.

At a high level, our proof works as follows. We assume for contradiction that $Q < \frac{n\ln(1{=})}{20000\ ^2 K}$. Using the assumption $\Pr_{H_1}[\mathrm{val}_K^{H_1}(T) \geq \mathrm{val}_K^{H_1}([n]) - \ ] \geq 1 - $, we can first prove that:

$$\Pr_{H_{j_0}}[\mathrm{val}_K^{H_1}(T) \geq \mathrm{val}_K^{H_1}([n]) - \ ] \geq \frac{\sqrt{\ }}{4}: \tag{5}$$

We further observe that when $\mathrm{val}_K^{H_1}(T) \geq \mathrm{val}_K^{H_1}([n]) - $, T must consist of more than $\frac{3}{4}K$ arms from the first group; while when $\mathrm{val}_K^{H_{j_0}}(T) \geq \mathrm{val}_K^{H_{j_0}}([n]) - $, T must consist of more than $\frac{3}{4}K$ arms from the j_0-th group. Therefore the two events are mutually exclusive and we have: $\Pr_{H_{j_0}}\ \mathrm{val}_K^{H_{j_0}}(T) \geq \mathrm{val}_K^{H_{j_0}}([n]) - \ \leq 1 - \Pr_{H_{j_0}}\ \mathrm{val}_K^{H_1}(T) \geq \mathrm{val}_K^{H_1}([n]) - \ \leq 1 - \frac{\sqrt{\ }}{4} \leq 1 - 2$; where the last inequality is because of $< 0{:}01$. This contradicts the performance guarantees of Algorithm $\mathcal{A}$ and thus we conclude our proof.

The most technical part is to prove (5). To this end, we construct the likelihood ratio between events under the hypothesis H_1 and H_{j_0}. The intuition is that H_1 and H_{j_0} are similar, thus for any sampling outcomes y, the probability that H_1 generates y is close to H_{j_0}. Since $\mathcal{A}$ is deterministic, the sampling outcomes determine the next action and the final decision. Using this argument, we can show that if the event $\{\mathrm{val}_K^{H_1}(T) \geq \mathrm{val}_K^{H_1}([n]) - \ \}$ happens under H_1 with probability at least $1 - $, it would happen under H_{j_0} with a significant probability (i.e., at least $\frac{\sqrt{\ }}{4}$). The details are provided in the appendix.

The proof of Lemma 5.4 can be easily generalized to the case where $\mathcal{A}$ is randomized, which allows us to prove the following stronger lower bound statement. The proof of Theorem 5.5 is relegated to the appendix.

Theorem 5.5. *Fix real numbers* ; *such that* $0 <$; $\leq 0{:}01$*, and integers* $\mathsf{K}; \mathsf{n}$*; where* $\mathsf{K} \leq \mathsf{n}/2$*. Let* $\mathcal{A}$ *be a (possibly randomized) algorithm so that for any set of* n *arms with the mean* $_1; _2; \ldots; _n$*,*

- $\mathcal{A}$ *makes at most* Q *samples in expectation;*
- *With probability at least* $1 -$ *,* $\mathcal{A}$ *outputs a set* T *of size* K *with* $\mathsf{val}_K(\mathsf{T}) \geq \mathsf{val}_K([\mathsf{n}]) -$.

We have that $\mathsf{Q} = \Omega\left(\frac{n \ln(1/)}{^2 K}\right)$.

6 Experiments

In this experiment, we assume that arms follow independent Bernoulli distributions with different means. To make a fair comparison, we fix the total budget Q and compare our algorithm (OptMAI) with the uniform sampling strategy and two other state-of-the-art algorithms: SAR (Bubeck et al., 2013) and LUCB (Kalyanakrishnan et al., 2012), in terms of the aggregate regret in (1).

The implementation of our algorithm is slightly different from its description in Section 3. While we choose to present a variant of our implementation only because of its simplicity of exposition, we describe the full details of the differences as follows. It is easy to check that our implementation still meets the theoretical bound proved in Section 4.

First, observe that in OptMAI, Q is an upper bound of the number of samples, while $(1 - {}^{R})\mathsf{Q} < \mathsf{Q}$ is the actual number of samples used, where R is the total number of rounds run by the algorithm. To fully utilize the budget, we run OptMAI with a parameter slightly greater than Q to ensure that the actual number of samples roughly equals to (but no greater than) Q .

Second, in each round of QE or AR, when computing the empirical mean $\hat{}_i$, our implementation uses all the samples obtained for the i-th arm (i.e. including the samples from previous rounds). This will lead to be better empirical performance especially when the budget is very limited.

Third, in each round of OptMAI, the ratio of the number of samples between two consecutive rounds is set to be $= \mathsf{e}^{0:2} \cdot 0{:}75 \approx 0{:}91$. In the real implementation, one could treat this quantity as a tuning parameter to make the algorithm more flexible (as long as $\in (0{:}75; 1)$). In this experiment, we report the results for both $= 0{:}8$ and $= 0{:}9$. Based on our experimental results, one could simply set $= 0{:}8$, which will lead to reasonably good performance under different scenarios. We propose the following strategy to tune as a future work. In the first stage, we sample each arm for a few times and then use the empirical estimate of $_i$ to generate as much simulated data as we want. Then, we choose the best based on the simulated data. Finally, we apply the carefully tuned to the real data using the remaining budget.

6.1 Simulated Experiments

In our simulated experiment, the number of total arms is set to $\mathsf{n} = 1000$. We vary the total budget $\mathsf{Q} = 20\mathsf{n}; 50\mathsf{n}; 100\mathsf{n}$ and $\mathsf{K} = 10; 20; \ldots; 500$. We use different ways to generate $\{ _i\}_{i=1}^{n}$ and report the comparison results among different algorithms:

1. $_i \sim \mathrm{Unif}[0; 1]$: each $_i$ is uniformly distributed on $[0; 1]$ (see Figure 1(a) to Figure 1(b)).
2. $_i = 0{:}5/0{:}6$: $_i = 0{:}6$ for $\mathsf{i} = 1; \ldots; \mathsf{K}$ and $_i = 0{:}5$ for $\mathsf{i} = \mathsf{K} + 1; \ldots; \mathsf{n}$. We note that such a two level setting of $_i$ is more challenging for the selection of top-K arms (see Figure 1(d) to Figure 1(f)).

It can be seen from Figure 1 that the uniform sampling performs the worst and our method outperforms SAR and LUCB in most of the scenarios. We also observe that when K is large, the setting of $= 0{:}8$ (red line) outperforms that of $= 0{:}9$; while for small K, $= 0{:}9$ (blue line) is a better choice.

We also generate $_i$ from the truncated normal distribution and the Beta distribution and have similar observations. The comparison results are presented in the appendix due to space constraints.

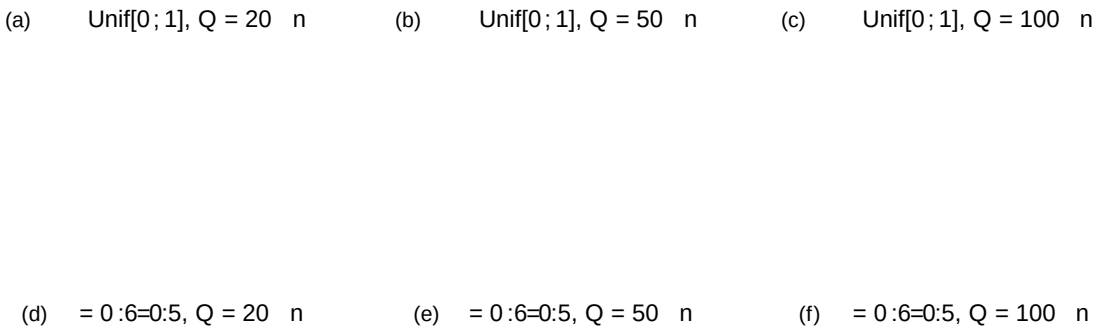

Figure 1: Performance comparison on simulated data.

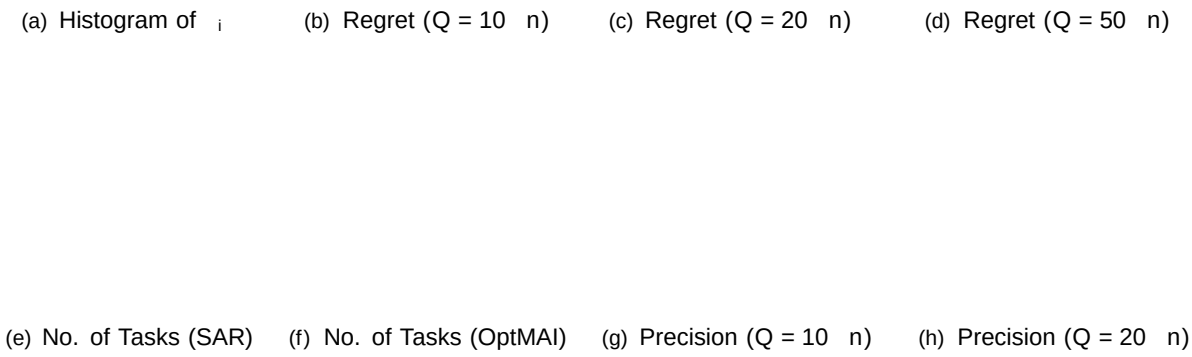

Figure 2: Performance comparison on the RTE data.

6.2 Real RTE Data

We generate from a real recognizing textual entailment (RTE) dataset (Section 4.3 in (Snow et al., 2008)). There are 800 task and each task is a sentence pair. Each sentence pair is presented to 10 di erent workers to acquire binary choices of whether the second hypothesis sentence can be inferred from the rst one. There are in total 164 di erent workers. Since there are true labels of tasks, we set each for the i-th worker to be his/her labeling accuracy. The histogram of i is presented in Figure 2(a). We vary the total budget