Adaptive Private-K-Selection with Adaptive K and Application to Multi-label PATE

Yuqing Zhu
UC Santa Barbara

Yu-Xiang Wang
UC Santa Barbara

Abstract

We provide an end-to-end Renyi DP based-framework for differentially private top-\(k\) selection. Unlike previous approaches, which require a data-independent choice on \(k\), we propose to privately release a data-dependent choice of \(k\) such that the gap between \(k\)-th and the \((k + 1)\)st “quality” is large. This is achieved by a novel application of the Report-Noisy-Max. Not only does this eliminate one hyperparameter, the adaptive choice of \(k\) also certifies the stability of the top-\(k\) indices in the unordered set so we can release them using a variant of propose-test-release (PTR) without adding noise. We show that our construction improves the privacy-utility trade-offs compared to the previous top-\(k\) selection algorithms theoretically and empirically. Additionally, we apply our algorithm to “Private Aggregation of Teacher Ensembles (PATE)” in multi-label classification tasks with a large number of labels and show that it leads to significant performance gains.

1 Introduction

The private top-\(k\) selection problem [Durfee and Rogers, 2019, Carvalho et al., 2020, Dwork et al., 2018, Hardt and Roth, 2013] is one of the most fundamental problems in privacy-preserving data analysis. For example, it is a key component in several more complicated differentially private tasks, including private model selection, heavy hitter estimation and dimension reduction. More recently, the private selection algorithm (Report-Noisy-Max) is combined with the “Private Aggregation of Teacher Ensembles (PATE)” [Papernot et al., 2017, 2018, Bassily et al., 2018] to build a knowledge trans-
approach to handle the case when there is a target \( k \) of interest. Empirical and analytical results demonstrate the utility improvements compared to the state-of-the-arts, which encouragingly suggests that using the PTR framework and Report-Noisy-Max with extensions tailored to our problem of interest. Our RDP analysis of RNM with other noise-adding mechanisms (e.g., Gaussian noise) is based on the proof technique of [Zhu and Wang, 2020] for analyzing SVT. Our approach may strike the readers as being very simple, but we emphasize that “constant matters in differential privacy” and the simplicity is precisely the reason why our method admits a tight privacy analysis. In our humble opinion all fundamental problems in DP should admit simple solutions and we are glad to have found one for private-k-selection.

2 Preliminary

In this work, we study the problem of differential private top-\( k \) selection in the user-counting setting\(^1\). Consider a dataset of \( n \) users is defined as \( D = \{x_1, ..., x_n\} \). We say that two dataset \( D \) and \( D' \) are neighboring, if they differ in any one user’s data, e.g. \( D = D' \cup \{x_i\} \). Assume a candidate set contains \( m \) candidates \( \{1, ..., m\} \). We consider the setting where a user can vote 1 for an arbitrary number of candidates, i.e. unrestricted sensitivity. One example for the unrestricted setting would be calculating the top-\( k \) popular places that users have visited. We use \( x_{j,i} \) to denote the voting of user \( i \), e.g., \( x_{j,i} = 1 \) indicates the \( i \)-th user vote 1 for the \( j \)-th candidate. Let \( h_j(D) \in \mathbb{N} \) denote the number of users that have element \( j \in [m] \), i.e. \( h_j(D) = \sum_{i=1}^{n} \mathbb{I}(x_{j,i} = 1) \) (we will drop \( D \) when it is clear from context).

We then sort the counts and denote \( h_{(1)}(D) \geq ... \geq h_{(m)}(D) \) as the sorted counts where \( i_{(1)}, ..., i_{(m)} \in [m] \) are the corresponding candidates. Our goal is to design a differentially private mechanism that outputs the unordered set \( \{i_{(1)}, ..., i_{(k)}\} \) which \( k \) is chosen adaptively to private data itself. Formally, the algorithm returns a \( m \)-dim indicator \( \mathbb{I}(D) \in \{0, 1\}^m \), where \( \mathbb{I}_j = 1 \) if \( j \in \{i_{(1)}, ..., i_{(k)}\} \), otherwise \( \mathbb{I}_j = 0 \).

**Symbols and notations.** Throughout the paper, we will use the standard notations for probability, e.g., \( \Pr[\cdot] \) for probability, \( \rho[\cdot] \) for density, \( \mathbb{E} \) for expectation. \( \epsilon, \delta \) are reserved for privacy loss parameters, and \( \alpha \) the order of Renyi DP. We now introduce the definition of differential privacy.

**Definition 1** (Differential privacy [Dwork et al., 2006]). A randomized algorithm \( M \) is \((\epsilon, \delta)\)-differential private.
if for neighboring dataset \( D \) and \( D' \) and all possible outcome sets \( O \subseteq \text{Range}(\mathcal{M}) \):

\[
\Pr[\mathcal{M}(D) \in O] \leq e^\epsilon \Pr[\mathcal{M}(D') \in O] + \delta
\]

Differential Privacy ensures that an adversary could not reliably infer whether one particular individual is in the dataset or not, even with arbitrary side-information.

**Definition 2** (Renyi DP [Mironov, 2017]). We say a randomized algorithm \( \mathcal{M} \) is \((\alpha, \epsilon, \chi(\alpha))\)-RDP with order \( \alpha \geq 1 \) if for neighboring datasets \( D, D' \)

\[
\Delta_{\alpha}(\mathcal{M}(D)||\mathcal{M}(D')) := \frac{1}{\alpha - 1} \log \mathbb{E}_{\alpha \sim \mathcal{M}(D')} \left[ \frac{\Pr[\mathcal{M}(D) = \alpha]}{\Pr[\mathcal{M}(D') = \alpha]} \right]^\alpha \leq \chi(\alpha).
\]

At the limit of \( \alpha \to \infty \), RDP reduces to \((\epsilon, 0)\)-DP. If \( \chi(\alpha) \leq \rho \alpha \) for all \( \alpha \), then we say that the algorithm satisfies \( \rho \)-zCDP [Bun and Steinke, 2016]. This more-fine-grained description often allows for a tighter \((\epsilon, \delta)\)-DP over compositions compared to the strong composition theorem in Kairouz et al. [2015]. Therefore, we choose to formulate the privacy guarantee of our algorithms under the RDP framework. Here, we introduce two properties of RDP that we will use.

**Lemma 3** (Adaptive composition). \( \chi(\mathcal{M}_1, \mathcal{M}_2) = \chi(\mathcal{M}_1) + \chi(\mathcal{M}_2) \).

**Lemma 4** (From RDP to DP). If a randomized algorithm \( \mathcal{M} \) satisfies \((\alpha, \epsilon(\alpha))\)-RDP, then \( \mathcal{M} \) also satisfies \((\epsilon(\alpha) + 1/\alpha, \delta)\)-DP for any \( \delta \in (0, 1) \).

Next, we will introduce the notion of approximate RDP, which generalizes approximate zCDP [Bun and Steinke, 2016].

**Definition 5** (Approximate RDP / zCDP). We say a randomized algorithm \( \mathcal{M} \) is \( \delta \)-approximately \((\alpha, \epsilon(\alpha))\)-RDP with order \( \alpha \geq 1 \), if for all neighboring dataset \( D \) and \( D' \), there exist events \( E \) (depending on \( \mathcal{M}(D) \) and \( E' \) (depending on \( \mathcal{M}(D') \)) such that \( \Pr[E] \geq 1 - \delta \) and \( \Pr[E'] \geq 1 - \delta \), and \( \forall \alpha \geq 1 \), we have \( \Delta_{\alpha}(\mathcal{M}(D)||\mathcal{M}(D')|E \cap E') \leq \epsilon(\alpha) \). When \( \epsilon(\alpha) \leq \alpha \rho \) for \( \alpha \geq 1 \) then \( \mathcal{M} \) satisfies \( \delta \)-approximate \( \rho \)-zCDP.

This notion preserves all the properties as approximate zCDP [Bun and Steinke, 2016]. The reason for rephrasing it under the RDP framework is that some of our proposed algorithms satisfy tighter RDP guarantees (compared to its zCDP version) while others satisfy RDP conditioning on certain high probability events. Similar conversion and composition rules of approximate-RDP are deferred to the appendix.

Many differentially private algorithms, including output perturbation, enable DP working by calibrating noise using the sensitivity. We start by defining the local and global sensitivity.

**Definition 6** (Local / Global sensitivity). The local sensitivity of \( f \) with the dataset \( D \) is defined as \( LS_f(D) = \sup_{D' \sim D} ||f(D) - f(D')|| \) and the global sensitivity of \( f \) is \( GS_f := \sup_D LS_f(D) \).

The norm \( || \cdot || \) could be any vector \( \ell_p \) norm, and the choice on \( \ell_p \) depends on which kind of noise we use, e.g., we calibrate Gaussian noise for Gaussian mechanism using \( \ell_2 \) norm.

**Motivation of an adaptive \( k \)**. Recent work [Carvalho et al., 2020, Durfee and Rogers, 2019, Gillenwater et al., 2022] make use of structures in the top-\( k \) counts, showing that large gaps improve the performance of the private top-\( k \) mechanisms. This leads to one natural question — can’t we just set a \( k \), such that there exists a large gap between the \( k \) and the \( (k + 1) \)th vote? Indeed, exploiting such the largest eigengap information is already a standard heuristic in selecting the number of principal components in PCA. Our result shows that if there is a large gap between \( k \)-th and the \( (k + 1) \)th, we can return the top \( k \) set with only two times privacy budget instead of \( k \) times. Moreover, even if want a pre-defined \( k \) and there is a large gap at \( (k - 3) \), then we can release the top \( (k - 3) \) with two times the budget then release the remaining using the exponential mechanism with 3 times the budget. Our motivation is to adapt to these large-margin structures.

## 3 Methods

We now present our main algorithms for data-adaptive top-\( k \) selection. Section 3.1 describes a simple algorithm that privately selects parameter \( k \in [m] \) such that it maximizes the gap \( h(k) - h(k + 1) \). Section 3.2 presents a propose-test-release style algorithm called STABLETOPK. It first privately selects \( k \in [m] \) such that it maximizes gap, then releases the top-\( k \) index set whenever the gap at the chosen \( k \) is large. Section 3.3 demonstrates how STABLETOPK can be used for the fixed \( k \) setting, where the algorithm takes \( k \) as an input and is required to return exactly \( k \) indices.

### 3.1 Choose a \( k \) privately

Recall that the goal is to choose \( k \) that approximately maximizes the gap \( h(k) - h(k + 1) \). Our idea of choosing \( k \) uses off-the-shelf differentially private (Top-1) selection algorithms. Any private selection algorithm will work, but for simplicity we focus on the exponential mechanism [McSherry and Talwar, 2007], which is recently shown to admit a Report-Noise-Max style implementation and a more refined privacy analysis.
Algorithm 1: Regularized Large Gap

1: **Input** Histogram $h$, regularizer $r: |m - 1| \to \mathbb{R}$; DP parameter $\epsilon$.
2: Sort $h$ into a descending order $h(1), h(2), ..., h(m)$.
3: **Return** $\text{Argmax}_{j \in [m - 1]} \{h(j) - h(j + 1) + r(j) + \text{Gumbel}(\frac{2}{\epsilon})\}$. 

via a “Bounded Range” property [Durfee and Rogers, 2019].

The pseudo-code is given in Algorithm 1. Readers may notice that it also takes a regularizer $r$. The choice of $r$ can be arbitrary and can be used to encode additional public information that the data analyst supplies such as hard constraints or priors that describe the “ball park” of interest.

**Proposition 7.** Algorithm 1 satisfies (pure)-$\epsilon$-DP, $\epsilon^2/8$-zCDP and and $(\alpha, c(\alpha))$-RDP with

$$\epsilon(\alpha) := \min \left\{ \frac{\alpha^2}{8}, \frac{1}{\alpha - 1} \log \left( \frac{\sinh(\alpha) - \sinh((\alpha - 1)\epsilon)}{\sinh(\epsilon)} \right) \right\}.$$ 

**Proof.** As we are applying the exponential mechanism off-the-shelf, it suffices to analyze the sensitivity of the utility function $u(j) := h(j) - h(j + 1) + r(j)$. Let $u, u'$ be the utility function of two neighboring dataset (with histograms $h, h'$). For any $j$

$$|u(j) - u'(j)| = |(h(j) - h(j + 1)) - (h'(j) - h'(j + 1))| \leq 1.$$ 

The inequality can be seen by discussing the two cases: “adding” and “removing” separately. If we add one data point, it may only increase $h(j)$ and $h(j + 1)$ by 1. Similarly if we remove one data point it may only decrease $h(j)$ and $h(j + 1)$. In both cases, the change of the gap is at most 1. The pure-DP bound follows from McSherry and Talwar [2007], the zCDP bound follows from Cesar and Rogers [2021, Lemma 17] and the RDP bound is due to Bun and Steinke [2016, Lemma 4].

Algorithm 1 is exponentially more likely to return a $k$ that has a larger gap than $k$ that has a small gap. In our experiments, we find that the tighter zCDP analysis gives EM an advantage over other alternatives including the exponential noise and Laplace noise versions of RNM [Ding et al., 2021]. For this reason, discussion of these other selection procedures are given in the appendix.

**Gaussian-RNM.** One may ask a natural question whether one can use more concentrated noise such as Gaussian noise to instantiate RNM. Using the techniques from Zhu and Wang [2020], we prove the following theorem about such generalized RNMs.

**Theorem 8.** Let $M_g$ denote any noise-adding mechanism that satisfies $\epsilon_g(\alpha)$-RDP for a scalar function $f$ with global sensitivity $2$. Assume Report Noisy Max adds the same magnitude of noise to each coordinate, then the algorithm obeys $\epsilon(\alpha)(|M(D)||M(D')|) \leq \epsilon_g(\alpha) + \frac{\log m}{\alpha - 1}$.

In particular, we introduce RNM-Gaussian as an alternative to RNM-Laplace with Gaussian noise.

**Corollary 9 (RNM-Gaussian).** RNM-Gaussian (the second line in Algorithm 2) with Gaussian noise $N(0, \sigma^2)$ satisfies $\left( \frac{2\epsilon}{\sigma^2} + \frac{\log m}{\alpha - 1} \right)$-RDP.

We defer the comparison between RNM-variants in the appendix, suggesting that RNM-Gaussian is better than RNM-Laplace in certain regime (e.g., $m$ is not too large). However, RNM-Gumbel will dominate both of them over compositions.

**How to handle unknown domain / unlimited domain?** In TopK selection problems, it is usually desirable to be able to handle an unbounded $m$ in an unknown domain [Durfee and Rogers, 2019, Carvalho et al., 2020]. Our method handles it naturally by taking the regularizer $r$ to be a constraint that restricts our choices to $j \in \{1, 2, ..., k\}$ with an arbitrary $k \ll m$. The issue of candidates moving inside and outside the top $k$ is naturally handled by the selection of a stable $k$ within $\{1, 2, ..., \bar{k}\}$. This simultaneously improves the RDP bound and the utility bound for RNM-Gaussian by replacing $m$ with $\bar{k}$.

### 3.2 Stable Top-k selection with an adaptive $k$

Once $k$ is determined, the next step is to privately release the top $k$ index set. Different from existing methods that select the top $k$ by iteratively calling exponential mechanisms for $k$ times, we propose a new approach that release the unordered indices of the top $k$ at one shot using a propose-test-release (PTR) [Dwork and Lei, 2009] style algorithm. The query of interest is the indicator vector $I_k(D) \in \{0, 1\}^m$ satisfying

$$[I_k(D)]_j = \begin{cases} 1 & \text{if } j \in \text{TopK} \\ 0 & \text{otherwise.} \end{cases}$$

The indicator has a global L2 sensitivity of $\sqrt{2k}$, as there are at most $k$ positions are 1 in $I(D)$ and $\mathbb{E}I(D)$. It could appear to be a silly idea to apply Gaussian mechanism, because a naive application would require adding noise with scale $\approx \mathcal{N}(0, \sqrt{2k}I_m)$, rendering an almost useless release. Luckily, the problem happens to be one where the global sensitivity is way too conservative, and one can get away with adding a much smaller noise in a typical dataset, as the following lemma shows.

**Lemma 10 (Local sensitivity of the gap).** Denote $q_k(D) := h(k)(D) - h(k + 1)(D)$ as the gap between the $k$-
Algorithm 2 StableTopK: Private k selection with an adaptive chosen k

1: **Input** Histogram $h$ and approximate zCDP budget parameters $\delta, \rho$.
2: Set $k$ by invoking Algorithm 1 with $\epsilon = 2\sqrt{r}$ (and arbitrary $r$).
3: Set $q_k = h(k) - h(k+1)$ and $\sigma = \sqrt{1/r}$.
4: Construct a high-probability lower bound $\hat{q}_k = \max\{1, q_k\} + \mathcal{N}(0, \sigma^2) - \sigma \sqrt{2\log(1/h)}$.
5: if $\hat{q}_k > 1$ then
6: Return $i(1), \ldots, i(k)$
7: else
8: Return $\perp$
9: end if

and the $k+1$-th largest count. The local $\ell_2$ sensitivity of $q_k$ is $0$ if $q_k(D) > 1$.

**Proof.** Fix $k$. If we are adding, then it could increase $h(k+1)(D)$ by at most 1 and may not decrease $h(k)(D)$.
If we are removing, then it could decrease $h(k)(D)$ by at most 1 and may not increase $h(k+1)(D)$. In either case, if $q_k(D) > 1$, it implies that $h(k+1)(D') < h(k)(D')$, thus the set of the top $k$ indices remains unchanged. $\square$

Using the PTR approach, if we differentially privately test that the local sensitivity is indeed 0, then we can get away with returning $\perp(D)$ as is without adding any noise. Notably, this approach avoids composition over $k$ rounds and could lead to orders of magnitude improvements over the iterative EM baseline when $k$ is large. A pseudocode of our proposed mechanism is given in Algorithm 2.

**Theorem 11.** Algorithm 2 satisfies $\delta_1$-approximated-$\rho$-zCDP and $(\rho + \sqrt{2}\log(1/\delta), \delta + \delta_1)$-DP for any $\delta \geq 0$. Moreover, if the chosen $k$ satisfies $q_k > 1 + 2\sqrt{2\log(1/\delta_1)/\rho}$, the algorithm returns the correct top-$k$ set with probability $1 - \delta$.

**Proof.** The mechanism is a composition of Algorithm 1 (by the choice of parameter, it satisfies $\rho/2$-zCDP) and an application of PTR which is shown to satisfy $\delta_1$-approximate $\rho/2$-zCDP in Lemma 18 in the appendix. The stated result is obtained by the composition of approximate zCDP and its conversion to $(\epsilon, \delta)$-DP. Finally, the utility statement follows straightforwardly from the standard subgaussian tail bound. $\square$

**Utility comparison.** The theorem shows that our algorithm returns the correct Top-$k$ index with high probability if the gap $q_k$ is $O(\sqrt{2\log(1/\delta_1)/\rho})$. In comparison, the iterative EM algorithm, or its limited domain (LD) variant [Durfee and Rogers, 2019] requires the gap to be on the order of $O(\frac{2\log(1/\delta_1)}{\rho})$ — a factor of $\sqrt{k\log(1/\delta_1)}$ worse than our results. Comparing to the Top Stable procedure (TS) [Carvalho et al., 2020], which is similar to our method, but uses SVT instead of EM for selection; under the same condition (by Theorem 4.1 in their paper) TS requires the gap to be $\log(1/\delta_1)/\sqrt{\rho}$, which is a factor of $\sqrt{\log(1/\delta)}$ larger than our results.

**Connection to distance to instability framework**
Our algorithm has a nice connection with the distance to instability framework [Thakurta and Smith, 2013]. Similar to the idea of using gap information to upper bound the local sensitivity, we can define the Dist2Instability function to be $\max\{0, h(k)(D) - h(k+1)(D) - 1\}$ and test whether it is 0 using Laplace mechanism. Our PTR-Gaussian algorithm can be thought of as an extension of the distance to instability framework with Gaussian noise, which is of independent interest.

**Why not smooth sensitivity?** A popular alternative to PTR for such tasks of data-adaptive DP algorithm is the smooth sensitivity framework [Nissim et al., 2007], which requires constructing an exponentially smoothed upper bound of the local sensitivity and add noise that satisfy certain “dilation” and “shift” properties. Our problem does have an efficient smooth sensitivity calculation, however, we find that the “dilation” and “shift” properties of typical noise distributions (including more recent ones such as those proposed in Bun and Steinke [2019]) deteriorate exponentially as dimensionality gets large; making it infeasible for releasing an extremely high-dimensional vector in $\{0, 1\}^m$.

### 3.3 Stable private k-selection with a fixed k

In many scenarios, $k$ is a parameter chosen by the data analyst who expect the algorithm to return exactly $k$ elements. In this situation, there might not be a large gap at $k$. In this section, we show that one can still benefit from a large gap in this setting if there exists one in the the neighborhood of the chosen $k$.

We introduce StableTopK with a fixed $k$ (Algorithm 3) which take as input a histogram $h$, parameter $k$, regularizer parameter $\lambda$, and approximate zCDP parameter $\delta_1$ and $\rho$.

Ideally, we hope to find a $\tilde{k}$ such that we see a sudden drop at the $\tilde{k}$-th position and $\tilde{k}$ is closed to the input $k$. Therefore, we introduce a regularizer term $\lambda |j - k|$ in Step 2. Then we apply PTR-Gaussian (Algorithm 4) to privately release the top-$k$ elements. If $\tilde{k} < k$, we can optionally use exponential mechanism ([McSherry and Talwar, 2007]) to privately select top-$(k - \tilde{k})$ elements in a peeling manner. Similarly, if $\tilde{k} > k$, we can apply...
Algorithm 3 StableTopK with fixed $k$: Private Top-$k$ selection with a fixed $k$ input

1: **Input** Histogram $h$, parameter $k$, regularizer weight $\lambda$, approx zCDP parameter $\delta_t, \rho$.
2: Set $r(j) = -\lambda|j-k|$.
3: Set $\epsilon_{EM} = 2\sqrt{\rho}$.
4: Set $S$ as the output of Algorithm 2, instantiated with $(h, \delta_t, \rho/2)$ and regularizer $r$.
5: if $S = \perp$, **Return** result of Top-$k$ EM on $h$ with total pure-DP budget $\epsilon_{EM}$.
6: if $\tilde{k} = k$, **Return** $S$.
7: **elif** $k > \tilde{k}$, **Return** result of Top-$k$ EM on $h_{i(t)}, ..., h_{i(k)}$ with budget $\epsilon_{EM}$.
8: **else** **Return** $\{i_1, ..., i_{\tilde{k}}\} \cup$ result of Top-$(k-\tilde{k})$ EM on $h$ with budget $\epsilon_{EM}$.

Example 3 (PATE with multi-class classification tasks [Papernot et al., 2018]). For each unlabeled data $x$ from the public domain, let $f_j(x) \in [c]$ denote the $j$-th teacher model’s prediction and $n_i$ denotes the vote count for the $i$-th class (i.e., $n_i := \sum_j |f_j(x) = i|$). PATE framework labels $x$ by $M_{PATE}(x) = \text{argmax}_i (n_i(x) + \mathcal{N}(0, \sigma^2))$. $M_{PATE}$ guarantees $(\alpha, \alpha/\sigma^2)$-RDP for each labeling query.

Unfortunately, the current PATE framework only supports the multi-class classification tasks instead of the generalized multi-label classification tasks, while the latter plays an essential role in private language model training (e.g., tag classification). The reasons are two-fold: first, the label space is large and each teacher in principle could vote for all labels (i.e., the global sensitivity grows linearly with the label space), thus preventing a practical privacy-utility tradeoff using Gaussian mechanism. Second, previous private top-$k$ selection algorithms do not target multiple releases of private queries; thus, there is a lack of a tight private accountant. Our algorithm naturally narrows this gap by providing an end-to-end RDP framework that enables a sharper composition. Moreover, an adaptively chosen $k$ is indeed favorable by PATE, as the number of ground-truth labels can be different across different unlabeled data. We provide one example of applying Algorithm 2 to solve multi-label classification tasks.

Example 4 (PATE with multi-label classification tasks). For each unlabeled data $x$ from the public domain, let $f_j(x) \in \{0,1\}^c$ denote the $j$-th teacher model’s prediction and $n_i$ denotes the vote count for the $i$-th class (i.e., $n_i := \sum_j |f_j(x)|$). PATE framework labels $x$ by $M_{PATE}(x) = \text{Algorithm 2}$. $M_{PATE}$ answers $T$ labeling queries guarantees $T\delta_t$-approximated-$p$-zCDP.

4 Experiment

EXP1: Evaluations of $k$-selection with a fixed $k$. In Exp 1, we compare our STABLETOPK with recent advances (TS[Carvalho et al., 2020] and the Limited Domain(LD) [Durfee and Rogers, 2019]) for private top-$k$ section algorithms. We replicate the experimental setups from [Carvalho et al., 2020], which contains two location-based check-ins datasets Foursquare [Yang et al., 2014] and BrightKite [Cho et al., 2011]. BrightKite contains over 100000 users and 1280000 candidates. Foursquare contains 2293 users with over 100000 candidates. We assume each
user gives at most one count when she visited a certain location. The goal is to select the top-$k$ most visited locations, where $k$ is chosen from $\{3, 10, 50\}$.

**Comparison Metrics and Settings** Similar to [Carvalho et al., 2020], we consider the proportion of true top-$k$ metric, which evaluates the number of true top-$k$ elements returned divided by $k$. For privacy budgets, we set $\delta = \frac{1}{n}$ and consider $\epsilon$ being chosen from $\{0.4, 0.8, 1.0\}$. In the calibration, we split half of $\delta$ as the failure probability $\delta_t$. Then, we use the RDP to $(\epsilon, \delta)$-DP conversion rule to calibrate $\sigma$ using the remaining privacy budget $(\epsilon, \delta - \delta_t)$. For simplicity, we use $r = 0$ in STABLETopK.

For TS and LD, we report their results from Carvalho et al. [2020].

**Observation** By increasing privacy budget from $\epsilon = 0.4$ to $\epsilon = 1.0$, the “accuracy” increases for all algorithms. Moreover, our STABLETopK consistently outperforms TS, LD in the specific settings we consider.

**EXP2: Multiple top-$k$ queries** The behaviors of top-$k$ mechanisms can be varied for different data distribution, $k$ and privacy budgets. To study their behaviors, we design two groups of experiments — one with a fixed $k$ but various data distribution and another with a range of $k$.

We first consider the case when $k$ is fixed with an instantiation in releasing daily top-$k$ states that has the largest Covid-19 cases. We will use the United States Covid-19 Cases by State from 2020-03-12 to 2020-05-12 and assume one person can contribute at most one case on the daily case report.

**Baselines and Metrics** TS and LD are two baselines. As the CDP/RDP analysis of both TS and LD is unknown, we use advanced composition to allocate the privacy budget $(\epsilon, \delta)$ over $T$ queries.

Another baseline we will use is the exponential mechanism EM-CDP. The exponential mechanism admits a tighter CDP analysis due to its bounded range property. We will add Gumbel noise to each count and report the indices with the top-$k$ highest noisy counts. In the experiment, we average the recall of the top-$k$ set over a fixed time interval (e.g., 10 days) and repeat each experiment for 100 trials.

In Figure 1(a), we consider $k = 15$ and $(0.1, 10^{-6})$-DP instances of EM-CDP, TS, LD and our STABLETopK(fixed K) For each mechanism, we first calibrate their noise scale such that the composition over 10 days satisfy $(0.1, 10^{-6})$-DP. We then simulate four groups of the time interval: Day 1-10, Day 11-20, Day 21-30 and Day 31-40 such that composition is the same but the distribution of daily covid-19 cases is varied. Note that there was an exponential growth on the covid-19 cases between 2020-03-12 to 2020-05-12, which leads to an increasing gap between the $k$-th and the $k+1$-th count.

EM-CDP performs best in all time intervals, especially when there are small gaps between the vote counts (see Day 1-10 and Day 11-20). When the gap is small, both TS and StableTopK will likely fail on the stability test, which will result in a substitute of the exponential mechanism using half of the privacy budget. Therefore, both TS and StableTopK perform worse than EM-CDP. The number of indices returned by LD can be smaller than $k$, especially when there is no large gap among vote counts. Thus it obtains the worst recall rate over all intervals. When the gap is large, all mechanisms achieve better performance. StableTopK is still slightly worse than EM-CDP on Day 31-40 though the latter requires splitting the privacy budget into $k$ pieces. We conjecture this is because the $k$ we use is small, which diminished the effect of “unavoidable $O(\sqrt{k})$ dependence in $\epsilon$” in EM-CDP.

---

**Figure 1:** Figure 1(a) evaluates composed top-$k$ selection with varied data distribution. Figure 1(b) compares top-$k$ selection with different choice on $k$. 

---
Therefore, we next construct a synthetic example to investigate the effect on $k$. The synthetic histogram has 15000 bins, where all top $k$ bins have 700 counts, and the remaining 15000−$k$ bins have 0 counts. We range $k$ from 10 to 1500 with (0.15, 10−6)-DP instances of EM-CDP, TS and our StableTopK. The line in Figure 1(b) plots the mean recall rate (of answering one-time top-$k$ query) from 100 trials, and the shaded region spans with the standard deviation for each mechanism. StableTopK outperforms all mechanisms, especially when $k$ is large. This is because the utility of StableTopK is determined by the gap at the $k$-th position rather than how large a $k$ is. StableTopK is clearly better than TS when $k$ is large. We note that Lyu et al. [2017] has a similar observation — EM outperforms SVT in the non-interactive setting. Though EM-CDP admits a tight composition through CDP, its peeling procedure requires splitting its privacy budget into $k$ splits for each subroutine. Therefore, EM-CDP is worse than StableTopK when $k$ is sufficiently large.

**EXP3: Evaluation with multi-label classification tasks.** CelebA [Liu et al., 2015] is a large-scale face attribute dataset with 220k celebrity images, each with 40 attribute annotations. To instantiate the PATE framework, we take the original training set as the private domain and split it into 800 teachers. Similar to the implementation from Zhu et al. [2020], we randomly pick 600 testing data to simulate unlabeled public data and using the remaining data for testing. We train each teacher model via a Resnet50m structure [He et al., 2016]. As there is no strict restriction on an exact $k$ output, we apply a Gaussian variant of Algorithm 2 (i.e., replace the second step in Algorithm 2 with RNM-Gaussian) with noisy parameters $\delta_t = 10^{-9}$, $\sigma_1 = 50, \sigma = 60$. $\sigma_1$ is used in RNM-Gaussian. Our result is compared to two baselines: PATE [Papernot et al., 2018] and PATE-τ [Zhu et al., 2020]. In PATE, the global sensitivity is 40, as each teacher can vote for all attributes. To limit the global sensitivity, PATE-τ applies a $\tau$-approximation by restricting each teacher’s vote that no more than $\tau$ attributes or contributions will be averaged to $\tau$. We remark that though the $\tau$ approximation approach significantly reduces the global sensitivity, the choice on $\tau$ shall not be data-dependent. In Table 2, we align the accuracy of three DP approaches and compare their accuracy at the test set. We report the privacy cost based on the composition of over 600 labeling queries from the public domain. For StableTopK, the reported $\epsilon$ is based on the RDP to DP conversion rule using $\delta = 10^{-6} – 600 \times 10^{-9}$. Each experiment is repeated five times. Our StableTopK (adaptive K) algorithm saves half of the privacy cost compared to PATE-τ while maintaining the same accuracy.

### 5 Conclusion

To conclude, we develop an efficient private top-$k$ algorithm with an end-to-end RDP analysis. We generalize the Report-Noisey-Max algorithm, the propose-test-release framework and the distance-to-instability framework with Gaussian noise and formal RDP analysis. In the downstream task, we show our algorithms improve the performance of the model-agnostic framework with multi-label classification. We hope this work will spark more practical applications of private selection algorithms.

**Acknowledgments**

The work was partially supported by NSF Award # 2048091, a Google Research Scholar Award and a gift from NEC Labs. Yuqing was partially supported by a Google PhD Fellowship. The authors would like to thank the anonymous reviewers for helpful feedback, as well as Jinshuo Dong and Ryan Rogers to helpful
discussions related to the exponential mechanism.

References


Nicolas Papernot, Martín Abadi, Úlfar Erlingsson, Ian Goodfellow, and Kunal Talwar. Semi-supervised knowledge transfer for deep learning from private


Our algorithm addresses this issue by exploiting the connection to privacy in the following corollary.

Next, we work out the detailed calibration of PTR approaches using Laplace/Gaussian noise and provide their privacy guarantee in the following corollary.
Algorithm 4 Propose-test-release (PTR) with Gaussian Noise

1: **Input** Histogram $h$, noise parameter $\sigma_t$ and the privacy parameter $\delta_t$
2: Let $i_{[1]}, \ldots, i_{(k)}$ be the unordered indices of the sorted histogram.
3: Set the gap $q_k = h_{(k)} - h_{(k+1)}$
4: Propose a private lower bound of $q_k$: $\hat{q}_k = \max\{1, q_k\} + \mathcal{N}(0, \sigma_2^2) - \sigma_2\sqrt{2\log(1/\delta_t)}$
5: If $\hat{q}_k \leq 1$, Return $\bot$
6: Return $i_{(1)}, \ldots, i_{(k)}$

Algorithm 5 Propose-test-release (PTR) with Laplace Noise

1: **Input** Histogram $h$, noisy $\hat{q}_k$, privacy parameter $\delta_t, \epsilon$
2: Let $i_{[1]}, \ldots, i_{(k)}$ be the unordered indices of the sorted histogram.
3: Set the gap $q_k = h_{(k)} - h_{(k+1)}$
4: Propose a private lower bound of $q_k$: $\hat{q}_k = q_k + \text{Lap}(1/\epsilon) - \log(1/\delta_t)/\epsilon$
5: If $\hat{q}_k \leq 1$, Return $\bot$
6: Return $i_{(1)}, \ldots, i_{(k)}$

**Corollary 19** (Privacy guarantee of PTR variants). Algorithm 5 (PTR-Laplace) satisfies $(\epsilon, \delta_t)$-DP. Algorithm 4 (PTR-Gaussian) satisfies $\delta_t$-approximately-$(\alpha, \frac{\epsilon^2}{2})$-RDP.

The Laplace noise used in PTR-Laplace is heavy-tailed, which requires the threshold in Algorithm 5 to be $O(\log(1/\delta_t))$ in order to control the failure probability being bounded by $\delta_t$. In contrast, Algorithm 4 with Gaussian noise requires a much smaller threshold — $O(\sqrt{\log 1/\delta_t})$ due to its more concentrated noise.

**Theorem 20** (Accuracy comparison). For one-time DP top-k query, the minimum gap $h_{(k)} - h_{(k+1)}$ needed to output $k$ elements with probability at least $1 - \beta$ is $h_{(k)} - h_{(k+1)} \geq 1 + (\log 1/\delta + \log 1/\beta)/(\epsilon/4)$ for PTR-Gaussian while $h_{(k)} - h_{(k+1)} > 1 + \log(1/\beta)/\epsilon + \log(1/\beta)/\epsilon$ for PTR-Laplace.

**Proof.** With $q_k \geq 1 + \log(1/\delta_t)/\epsilon + \log(1/\beta)/\epsilon$, we have

$$\hat{q}_k \geq 1 + \log(1/\delta_t)/\epsilon + \log(1/\beta)/\epsilon + \text{Lap}(1/\epsilon) - \log(1/\delta_t)/\epsilon = \log(1/\beta)/\epsilon + 1 + \text{Lap}(1/\epsilon)$$

PTR-Laplace outputs $k$ elements only when $\hat{q}_k > 1$. Therefore, the failure probability is bounded by $\Pr[\text{Lap}(1/\epsilon) > \log(1/\beta)] = \beta$. \hfill $\square$

PTR-Laplace outperforms PTR-Gaussian for one-time query as we are using a loose calibration $\sigma_2 = \sqrt{2\log(1/25/\delta)/\epsilon}$. However, if we align the zCDP parameter (e.g., $z_{gap}(\alpha) = \frac{\alpha^2}{2}$), then $\sigma_2 = 1/\epsilon$ and gives us the minimum gap to be $1 + \frac{1}{2}\sqrt{2\log(1/\delta)} + \frac{1}{2}\sqrt{2\log(1/\beta)}$ for PTR-Gaussian. This explains why the Gaussian version of PTR is superior under composition.

### C Omitted Proofs

**Theorem 21** (Restatement of Theorem 8). Let $M_q$ denote any noise-adding mechanism that satisfies $\epsilon_q(\alpha)$-RDP for a scalar function $f$ with global sensitivity 2. Assume Report-Noisy-Max adds the same magnitude of noise to each coordinate, then the algorithm obeys $\epsilon_q(M(D)||M(D')) \leq \epsilon_q(\alpha) + \frac{\log m}{\alpha-1}$.

**Proof.** We use $i$ to denote any possible output of the Report-Noisy-Max $M(D)$. The Report-Noisy-Max aims to select an coordinate $i$ that maximizes $C_i$ in a privacy-preserving way, where $C_i$ denotes the difference between $h_{(i)}(D)$ and $h_{((i+1)}(D)$. Let $C'$ denote the vector of the difference when the database is $D'$. We will use the Lipschitz property: for all $j \in [m-1], 1 + C'_j \geq C_j$. This is because adding/removing one data point could at most change $C_j$ by 1 for all $j \in [m-1]$. Throughout the proof, we will use $p(r_i), p(r_j)$ to denote the pdf of $r_i$ and $r_j$, where $r_i$ denote the realized noise added to the $i$-th coordinate.
From the definition of Renyi DP, we have
\[
D_\alpha(M(D)||M(D')) = \frac{1}{\alpha-1} \log \mathbb{E}_{D \sim D} \left[ \frac{\Pr[M(D) = i]}{\Pr[M(D') = i]} \right] = \frac{1}{\alpha-1} \log \sum_{i=1}^{m} \frac{\Pr[M(D) = i]}{\Pr[M(D') = i]}^{\alpha-1}
\]

(1)

Our goal is to upper bound \((*) = \sum_{i=1}^{m} \Pr[M(D) = i] \) The probability of outputting \(i\) can be written explicitly as follows:
\[
\Pr[M(D) = i] = \int_{-\infty}^{\infty} \frac{p(r_i)}{p(r_i)} \mathbb{P}[C_i + r_i > \max_{j \in [m], j \neq i} \{C_j + r_j\}] dr_i
\]
\[
= \int_{-\infty}^{\infty} \frac{p(r_i - 2)}{p(r_i)} \mathbb{P}[C_i + r_i - 2 > \max_{j \in [m], j \neq i} \{C_j + r_j\}] dr_i
\]
\[
= \mathbb{E}_{r_i} \left[ \left( \frac{p(r_i - 2)}{p(r_i)} \right) \mathbb{P}[C_i + r_i - 2 > \max_{j \in [m], j \neq i} \{C_j + r_j\}] \right]
\]

In the first step, the probability of \(\Pr[C_i + r_i > \max_{j \in [m], j \neq i} \{C_j + r_j\}]\) is over the randomness in \(r_j\). Substituting the above expression to the definition of RDP and apply Jensen’s inequality
\[
(*) \leq \sum_{i=1}^{m} \mathbb{E}_{r_i} \left[ \frac{p(r_i - 2)}{p(r_i)} \mathbb{P}[C_i + r_i - 2 > \max_{j \in [m], j \neq i} \{C_j + r_j\}] \right] \alpha - 1 \cdot \Pr[C_i + r_i - 2 > \max_{j \in [m], j \neq i} \{C_j + r_j\}]
\]

We apply Jensen’s inequality to bivariate function \(f(x, y) = x^{\alpha} y^{1-\alpha}\), which is jointly convex on \(R^2_+\) for \(\alpha \in (1, +\infty)\).

The key of the analysis relying on bounding \((**) = \left( \frac{\Pr[C_i + r_i - 2 > \max_{j \in [m], j \neq i} \{C_j + r_j\}]}{\Pr[C_i + r_i - 2 > \max_{j \in [m], j \neq i} \{C_j + r_j\}]}, \mathbb{P}[C_i + r_i - 2 > \max_{j \in [m], j \neq i} \{C_j + r_j\}] \right) \) Note that \(D'\) is constructed by adding or removing one user’s all predictions from \(D'\). In the worst-case scenario, we have \(C_j = C_j + 1\) for every \(j \in [m], j \neq i\). Based on the Lipschitz property, we have
\[
\Pr[C_i + r_i > \max_{j \in [m], j \neq i} \{C_j + r_j\}] \geq \Pr[C_i + r_i - 2 > \max_{j \in [m], j \neq i} \{C_j + r_j\}]
\]
which implies \((**) \leq 1\). Therefore, we have
\[
\epsilon_M(\alpha) \leq \frac{1}{\alpha-1} \log \sum_{i=1}^{m} \mathbb{E}_{r_i} \left[ \frac{p(r_i - 2)}{p(r_i)} \right] \leq \epsilon_g(\alpha) + \log(m) \frac{1}{\alpha-1}.
\]

\[\square\]

**Corollary 22 (Restatement of Corollary 9).** RNM-Gaussian (the second line in Algorithm 2) with Gaussian noise \(N(0, \sigma_1^2)\) satisfies \((\frac{2\alpha}{\sigma_1^2} + \log \frac{m}{\alpha-1})\)-RDP.

**Proof.** For a function \(f : D \rightarrow \mathbb{R}\) with L2 sensitivity 2, the RDP of Gaussian mechanism with Gaussian noise \(N(0, \sigma_1^2)\) satisfies \((\alpha, \frac{2\alpha}{\sigma_1^2})\)-RDP. We complete the proof by plugging in \(\epsilon_g(\alpha) = \frac{2\alpha}{\sigma_1^2}\) into Theorem 8. \[\square\]

**Lemma 23 (Restatement of Lemma 18).** Let \(\hat{q}_k\) obeys \(\epsilon_{gap}(\alpha)\)-RDP and \(\Pr[\hat{q}_k \geq q_k] \leq \delta_t\) (where the probability is only over the randomness in releasing \(\hat{q}_k\)). If \(\hat{q}_k\) passes the threshold check, the algorithm releases the set of top-k indices directly satisfies \(\delta_t(\alpha, \frac{\epsilon_{gap}(\alpha)}{\epsilon_{gap}(\alpha)})\)-RDP.

**Proof.** We start with the proof for \(\delta_t(\alpha, \frac{\epsilon_{gap}(\alpha)}{\epsilon_{gap}(\alpha)})\)-RDP. Denote \(M_1\) be the mechanism that releases the set of top-k indices directly (without adding noise) if \(\hat{q}_k\) passes the threshold check (\(\hat{q}_k > 1\)).

Then let us discuss the two cases of the neighboring pairs \(D, D'\).
(a) For neighboring datasets $D, D'$ where the Top-$k$ indices are the same, the possible outputs are therefore \{⊥, Top – k(D)\} for both $M_1(D), M_1(D')$. Notice that $|q_k(D) - q_k(D')| \leq 1$, thus in this case

$$\mathbb{D}_\alpha(M_1(D)||M_1(D')) = D_\alpha(1(\hat{q}_k(D) > 1)||1(\hat{q}_k(D') > 1)) \leq \mathbb{D}_\alpha(\hat{q}_k(D)||\hat{q}_k(D')) \leq \epsilon_{gap}(\alpha),$$

where the inequality follows from the information-processing inequality of the Renyi Divergence. Thus it trivially satisfies $\delta$-approximated-$(\alpha, \epsilon_{gap}(\alpha))$-RDP when we set $E$ to be the full set, i.e., $\Pr[E] = 1 \geq 1 - \delta$.

(b) For $D, D'$ where the Top-$k$ indices are different, then it implies that $q_k(D) \leq 1$ and $q_k(D') \leq 1$. In this case, we can construct $E$ to be the event where $\hat{q}_k \leq q_k$, i.e., the high-probability lower bound of $q_k$ is valid. Check that $\Pr[E] \geq 1 - \delta$ for any input dataset. Conditioning on $E$, $\hat{q}_k \leq q_k \leq 1$ for both $D, D'$, which implies that $\Pr[M_1(D) = \bot | E] = \Pr[M_1(D') = \bot | E] = 1$. Thus, trivially $\mathbb{D}_\alpha(M(D)||M(D')|E(D')) = 0$ for all $\alpha$. For this reason, it satisfies $\delta$-approximated-$(\alpha, \epsilon(\alpha))$-RDP for any function $\epsilon(\alpha) \geq 0$, which we instantiate it to be $\epsilon_{gap}(\alpha)$. 

\[\square\]