RRTxFM: Probabilistic Counting for Differentially Private Statistics

Saskia Nuñez von Voigt, Florian Tschorsch

▶ **To cite this version:**

Saskia Nuñez von Voigt, Florian Tschorsch. RRTxFM: Probabilistic Counting for Differentially Private Statistics. 18th Conference on e-Business, e-Services and e-Society (I3E), Sep 2019, Trondheim, Norway. pp.86-98, 10.1007/978-3-030-39634-3_9 . hal-03759107

HAL Id: hal-03759107
https://inria.hal.science/hal-03759107

Submitted on 24 Aug 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Distributed under a Creative Commons Attribution 4.0 International License

This document is the original author manuscript of a paper submitted to an IFIP conference proceedings or other IFIP publication by Springer Nature. As such, there may be some differences in the official published version of the paper. Such differences, if any, are usually due to reformatting during preparation for publication or minor corrections made by the author(s) during final proofreading of the publication manuscript.

RRTxFM: Probabilistic Counting for Differentially Private Statistics

Saskia Nuñez von Voigt and Florian Tschorsch

Distributed Security Infrastructure Group, Technical University of Berlin,
Straße des 17. Juni 135, 10623 Berlin, Germany
{saskia.nunezvonvoigt,florian.tschorsch}@tu-berlin.de
www.dsi.tu-berlin.de

Abstract. Data minimization has become a paradigm to address privacy concerns when collecting and storing personal data. In this paper we present two new approaches, `RSTxFM` and `RRTxFM`, to estimate the cardinality of a dataset while ensuring differential privacy. We argue that privacy-preserving cardinality estimators are able to realize strong privacy requirements. Both approaches are based on a probabilistic counting algorithm which has a logarithmic space complexity. We combine this with a randomization technique to provide differential privacy. In our analysis, we detail the privacy and utility guarantees and expose the impact of the various parameters. Moreover, we discuss workforce analytics as application area where strong privacy is paramount.

Keywords: Probabilistic Counting · Differential Privacy · Randomized Response.

1 Introduction

For data analytics, one of the fundamental operations is to determine the number of distinct elements in a data stream. Due to their small memory footprint and low computational overhead, probabilistic counting algorithms like FM sketches [14], Count-Min sketches [7], and Bloom filters [5] are widely used to estimate the set cardinality efficiently. In fact, they are suitable to record and derive statistics for any categorical data.

Probabilistic counting algorithms can also be used as privacy-enhancing technology, for example, to count Tor users [23], to collect browser statistics [11], or to track users moving from one area to another [3]. In our work, we consider workforce analytics as running example to illustrate a setting, where privacy is crucial and where we have to deal with data integration and data collection at the same time.

Example 1. In recent years, workforce or human resource (HR) analytics is growing rapidly [1]. Workforce analytics combines data from different HR systems and collects additional HR data to understand interrelationships, to predict trends, and to give advice for future developments. The prime example is to predict employee turnover and to infer its reasons by using workforce analytics [13].

For simplicity, assume we are interested in determining the number of employees who work overtime. We use a counting sketch, e.g., an FM sketch, to record the IDs of employees who work overtime on a monthly basis. By merging the corresponding sketches a data analyst should be able to estimate the number of employees who work overtime over arbitrary time ranges but unable to identify individual employees in the sketch.

In Europe, processing HR data requires special protection and is allowed only under certain circumstances, which is even more strictly regulated since the introduction of the General Data Protection Regulation (GDPR). One way to mitigate the risk of data misuse is to anonymize the data. However, incidents in the past have shown that supposedly anonymized data can be deanoymized. In 2006, Netflix published an anonymous dataset of film reviews for research purposes. By linking the dataset to auxiliary information, e.g., the Internet Movie Database, it was possible to identify the majority of users [18]. This result shows that pseudonymity is not sufficient to protect privacy.

In this paper, we propose two new approaches, `RSTxFM` and `RRTxFM`, for differentially private statistics by using privacy-enhanced FM sketches. To this end, we collect and aggregate data in sketches at a central point after performing our algorithms. We generally consider the counted data to be ephemeral and only `RSTxFM` and `RRTxFM` sketches to be persistent. Moreover, we assume that an (hones-but-curious) adversary knows the probabilistic counting algorithm, the IDs of all users in the dataset, and has access to the sketches. Even when using additional means of protection as in [23], the *absence* of an ID, i.e., the ID has not been recorded, reveals sensitive information. That is, in our example the adversary could reveal that an employee does *not* work overtime, which might be used to identify "unmotivated" personnel. We tackle this problem by employing a randomization step before recording IDs in a sketch. We mitigate the risk of being identified, independently of whether the user is in the dataset or not, by guaranteeing ϵ-differential privacy [9].

In the privacy analysis and the empirical evaluation, we expose the impact of the various system parameters. In particular, we show that our approaches provide strong differential privacy guarantees ($\epsilon < 1$), while still being able to produce accurate estimations (error $< 10\%$). We also discuss the merits of our two approaches: While `RSTxFM` is able to provide accurate results for very small ϵ, it strictly requires adding additional perturbation. In contrast, `RRTxFM` also provides differential privacy without this perturbation, which makes it the preferred solution when aggregating sketches. Accordingly, the main contributions of our paper can be summarized as follows:

- We identify probabilistic counting as basis for differentially private cardinality estimation in Section 3.
- We quantify the privacy level and prove that our algorithms satisfy differential privacy in Section 4.
- We analyse the accuracy of `RSTxFM` and `RRTxFM` in Section 5. We compare it to related approaches and show that appropriate parameters can be found to adjust the trade-off between the privacy and accuracy.

2 Related Work

Privacy-preserving statistics often consider a centralized architecture. The data is stored at a central place and noise is added to the output according to a Laplace or exponential distribution to reduce the risk for an individual to be identified [10, 12]. This approach however does not protect from data breaches performed by external or internal adversaries. Our approach is based on so-called FM sketches [14], which already aggregate data to some extent and therefore reduce the risks of a data breach.

Probabilistic data structures are generally suitable for privacy-enhanced data analytics [4, 15] as they reduce the amount of personal data and inherently follow the privacy principle of data minimization. Obfuscation by hashing IDs and relying on the probabilistic nature of the data structures alone is not sufficient to guarantee the privacy of all users [8]. Additional means of protection are necessary. However, even by adding additional noise [21, 23], it may become evident that an ID is not present in the dataset. While in some scenarios this might be a reasonable assumption, in our example (see Example 1), we consider that the absence of an ID also leaks sensitive information.

A multitude of approaches address the issue by combining the randomized response technique (RRT) [24] with Bloom filters to conceal a user ID's absence [2, 3, 11, 17, 19, 22]. For example, with RAPPOR [11] Google collects data about the startpage of Chrome users. The response (i.e., the user's startpage) is mapped to a Bloom filter. By employing a two-step RRT, RAPPOR flips each bit with a given probability and provides privacy, even if an attacker links several reports from a single user. In general, the accuracy of a Bloom filter depends on the number of utilized hash functions and the size of the Bloom filter, which increases linearly with the expected number of IDs. On the other hand, cardinality estimators and FM sketches in particular, require significantly less space (growing logarithmically with the number of IDs), which makes them more suitable if the number of distinct IDs is unknown in advance.

Because of the output perturbation, RAPPOR needs a high sample size for accurate estimations [20]. In contrast, PRIVAPPROX [20] perturbs the input and therefore requires a smaller sample size when compared to RAPPOR. Regarding the perturbation technique the approach is very similar to our approaches. However, PRIVAPPROX is designed for stream analytics and does not fit well with existing data like in workforce analytics. Therefore we use FM sketches which can be used to further combine and aggregate individual datasets.

3 Differentially Private Cardinality Estimators

We use Probabilistic Counting with Stochastic Averaging (PCSA) [14] as basis to estimate the number of distinct user IDs in a dataset. Accordingly, the family of probabilistic counting algorithms also became known as cardinality estimators. To some extent, our findings are applicable to cardinality estimators in general.

PCSA uses m FM sketches (with $m \geq 1$) in parallel and two hash functions h_1 and h_2. A single FM sketch is a bit array $B = b_1, ...b_L$, of length $L \geq 1$, which

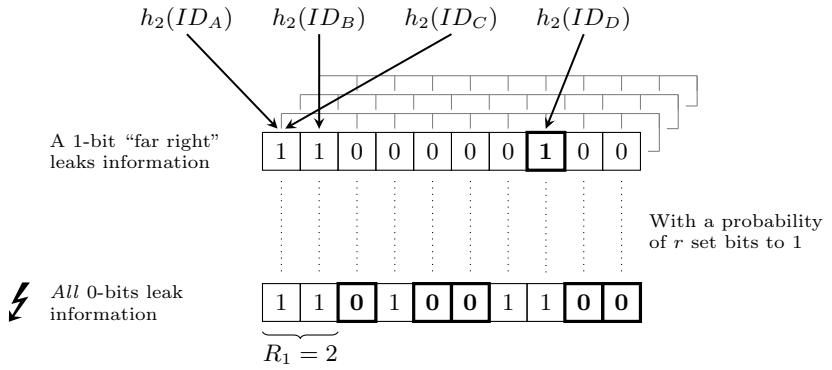

Fig. 1: Illustrating `PCSA(r)` as in [23] and revealing a privacy issue with 0-bits.

is initialized to zero. We count a user by hashing the ID and using the result to determine a bit position in one of the FM sketches that we set 1. More specifically, hash function h_1 is used to determine an FM sketch and h_2 to map the IDs to an index in the bit array. While h_1 is a uniformly distributed hash function, h_2 is a geometrically distributed hash function, which yields the probability $P(h_2(ID) = i) = 2^{-i}$ that a specific bit at index i is set. In practice, we also use a uniformly distributed hash function, inspect the binary representation of the hash value, and consider the least significant set bit's index as output. Assume for example that the binary representation of $h_2(ID_A) = [1001]_2$. The least significant set bit is $i = 1$ and therefore maps A's ID to the respective bit. We illustrate counting different IDs in Figure 1, where four distinct IDs (ID_A, ID_B, ID_C, and ID_D) are mapped on the first FM sketch.

Given the fact that h_2 is geometrically distributed, fewer IDs are mapped to higher indexes (right-hand side). In the worst case, only a single ID maps to a specific bit and an adversary can be sure that this ID was counted. To guarantee the privacy for all counted IDs, the authors of [23] introduce a perturbation technique. Each bit will be set with an additional probability r, which makes 1-bits "ambiguous" (cf. Figure 1). In the following, we will call this approach `PCSA(r)`. Note that if $r = 0$, the approach is identical to vanilla PCSA.

`PCSA(r)` also uses the number of consecutive 1-bits R_j to estimate the cardinality, but adapts the correction factor φ depending on r. The estimate C_{PCSA} is calculated with m FM sketches accordingly as

$$C_{\text{PCSA}} = \frac{m \cdot 2^{\sum_{j=1}^{m} R_j/m}}{\varphi(r)}. \tag{1}$$

When R_j is small, the estimation leads to inaccuracies. These can be mitigated to some extent by using a different estimation method based on "hit counting" [16] as long as the fraction of set bits (taking false positives into account) is below 30%, we consider the fraction k of 0-bits at the first bit position ($i = 1$) of each

sketch and calculate the cardinality as:

$$C_{\text{PCSA}} = (-2.0 \cdot m) \cdot \log\left(\frac{k}{m \cdot (1.0 - r)}\right).$$

While PCSA is generally well suited to estimate the cardinality, `PCSA(r)` also protects counted users/IDs. Unfortunately, the approach still leaks information: all 0-bits reveal that all respective user IDs have *not* been counted (cf. Figure 1). In our running example, we are interested to estimate the cardinality of all employees who work overtime. The absence of an employee ID indicates that this employee has not worked overtime, which reveals sensitive information. Accordingly, the privacy is not fully guaranteed.

In the following, we will present two approaches which tackle this privacy issue. Our general solution strategy is to induce "uncertainty" to the counting procedure with the goal to ensure privacy even if an adversary knows all user IDs in the dataset. To this end, we apply two randomization techniques to perturb the input. Our first algorithm `RSTxFM` uses *random sampling* to count only a sample of all IDs. Our second algorithm `RRTxFM` adopts the *randomized response technique* (RRT) to scramble the input in such a way that it contains true and false information. In both approaches, *each* bit in a sketch (0 *and* 1) yields plausible deniability as it remains unclear whether the answer is a result of randomization or truly corresponds to an ID. Later we will formalize this property and show that both approaches achieve differential privacy.

3.1 RSTxFM

In this approach, we randomly count a fraction p_1 of all IDs only. Let us assume we want to estimate the number of employees who work overtime as in Example 1. Moreover, assume that $\hat{p}$ is the fraction of employees who indeed work overtime, i.e., the set of IDs we are interested in. As shown in Figure 2, an employee working overtime is counted with a probability p_1. For counting user IDs, we use `PCSA(r)`. As a consequence, an adversary does not know which employees have been selected. A 0-bit can indicate that the corresponding employees did not work overtime or simply were not selected. We can still estimate the total cardinality C by evaluating the sketches according to Equation (1) and setting the result in proportion to p_1, which yields

$$C = \frac{C_{PCSA}}{p_1}. \tag{2}$$

3.2 RRTxFM

Our second approach follows the general idea of RRT [24], a method used in surveys to guarantee privacy. The data is perturbed in a way that a data collector cannot tell whether the answer contains true or false information. In recent years this method has been modified. We adopt the Forced Response Model [6]. The

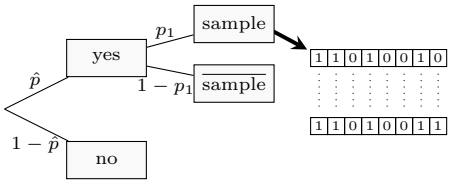

Fig. 2: Procedure of `RSTxFM`.

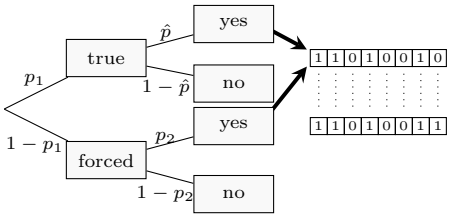

Fig. 3: Procedure of `RRTxFM`.

method can be best described by an example: Before answering a question an employee flips a coin. If the coin comes head the employee answers truthfully whether he works overtime. If the coin comes tail the employee's answer is forced by flipping another coin. For head the answer is "yes" (i.e., working overtime) and for tails "no" (i.e., not working overtime). In Figure 3, we sketch the procedure.

In order to control the impact of true and forced answers, we leave the parameters flexible and do not use a static coin flipping mechanism. With a probability p_1, we count the true fraction $\hat{p}$ of IDs we are interested in, e.g., employees working overtime. These IDs are mapped to a `PCSA(r)` sketch. With a probability $1 - p_1$ we use a forced answer. The forced answer is counted as well (i.e., "yes") with probability of p_2. With probabilities $p_1 = p_2 = 0.5$, `RRTxFM` is identical to the example using a coin flip to determine the input data.

With the probability tree in Figure 3 we can estimate the true fraction $\hat{p}$. Basically, there are two ways a bit can be set: by answering truthful and by a forced answer. The probability of getting a "yes" answer is $p_1 \cdot \hat{p} + (1 - p_1) \cdot p_2$. Setting the total number of "yes" responses C_{PCSA} equal to this probability and solving for $\hat{p}$, we can estimate the true cardinality C by calculating

$$C = \frac{\frac{C_{\text{PCSA}}}{N} - p_2 + p_1 \cdot p_2}{p_1} \cdot N. \tag{3}$$

4 Privacy Analysis

With our approaches we aim for satisfying the strict concept of ϵ-differential privacy introduced by Dwork et al. [9]. It guarantees privacy regardless of the amount of background knowledge of an adversary. Accordingly, a function f provides ϵ-differential privacy if all pairs of answers a_1 and a_2 and all $S \subseteq \text{Range}(f)$ satisfy

$$P[f(a_1) \in S] \leq \mathrm{e}^{\epsilon} P[f(a_2) \in S]. \tag{4}$$

A smaller ϵ generally yields a stronger privacy. For $\epsilon = 0$, the output of function f is the same independent of the input, i.e., it is irrelevant whether a_1 or a_2 is in the dataset. While $\epsilon = 0$ leads to the strongest privacy guarantees, it obviously cannot be used to obtain meaningful results. Finding a balance between the privacy level ϵ and the accuracy of results is necessary.

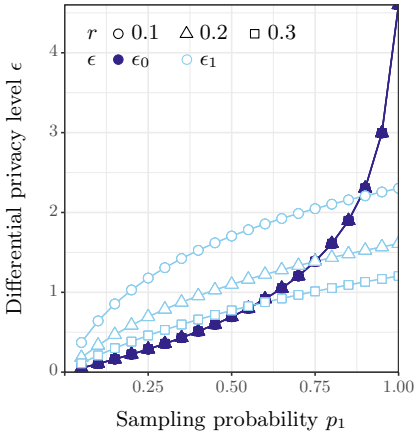

Fig. 4: Privacy level of `RSTxFM`.

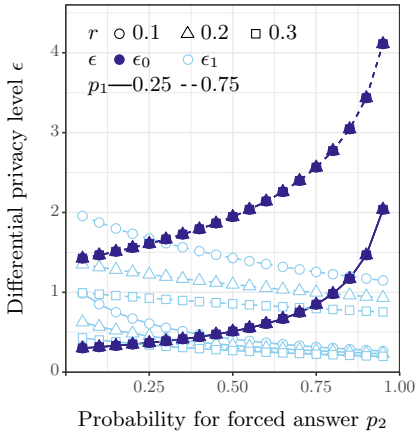

Fig. 5: Privacy level of `RRTxFM`.

Differential privacy expects the worst case [10]. For this reason we assume that an adversary knows all IDs and all algorithmic details, particularly the hash functions to map IDs to a sketch. The worst case is, as elaborated intuitively in the previous section, a bit where only a single ID is mapped to.

In the following, we show that our approaches are differentially private and satisfy Equation (4). Since IDs are mapped to a single bit only, the privacy level ϵ is independent of the number of sketches and we only need to derive ϵ-differential privacy for one sketch. Therefore, we have to distinguish two possible answers, 1 for a positive and 0 for a negative answer. Accordingly, we have to show

$$\epsilon_0 \geq \ln\left(\frac{P[f(0)=0]}{P[f(1)=0]}\right) \quad \text{and} \quad \epsilon_1 \geq \ln\left(\frac{P[f(1)=1]}{P[f(0)=1]}\right),$$

where ϵ_0 describes the privacy level for the absence and ϵ_1 for the presence of an ID. The differential privacy level ϵ is then given by the maximum of ϵ_0 and ϵ_1, i.e., $\epsilon = \max(\epsilon_0, \epsilon_1)$.

4.1 Privacy Level of RSTxFM

With `RSTxFM`, there are two reasons for setting a bit: either an ID has a certain property and is sampled with a probability p_1, or the bit is set by the perturbation technique of `PCSA(r)` with a probability r. We can use this observation to calculate the conditional probabilities $P[f(0)=0]$, $P[f(1)=0]$, $P[f(1)=1]$ and $P[f(0)=1]$ and derive ϵ_0 and ϵ_1 accordingly. That is,

$$\epsilon_0 = ln\left(\frac{1}{1-p_1}\right) \quad \text{and} \quad \epsilon_1 = ln\left(\frac{p_1 + (1-p_1)\cdot r}{r}\right).$$

First of all, please note that the privacy level depends on p_1 and r. Only for $r > 0$ and $p_1 \neq 1$, `RSTxFM` satisfies the definition of differential privacy. In

Figure 4, we plotted ϵ_0 and ϵ_1 with p_1 on the x-axis and varying values for r. The influence of p_1 and r is as expected. The probability r has no impact on ϵ_0. For high values of p_1, ϵ_0 increases quickly so that for $p_1 \to 1 : \epsilon_0 = \infty$. In contrast, ϵ_1 depends on both probabilities p_1 and r. Overall, for a decreasing p_1 and an increasing r, $\max(\epsilon_0, \epsilon_1)$ decreases and provides stronger privacy, respectively.

4.2 Privacy Level of RRTxFM

Proving ϵ-differential privacy for `RRTxFM` is equivalent to `RSTxFM`. First, we have to calculate the conditional probabilities, which now not only depend on p_1 and r but also on p_2, before we can derive ϵ_0 and ϵ_1. We obtain

$$\epsilon_0 = \ln\left(\frac{p_1 + (1-p_1)\cdot(1-p_2)}{(1-p_1)\cdot(1-p_2)}\right) \quad \text{and}$$
$$\epsilon_1 = \ln\left(\frac{p_1 + (1-p_1)\cdot p_2 + (1-p_1)\cdot(1-p_2)\cdot r}{p_1\cdot r + (1-p_1)\cdot p_2 + (1-p_1)\cdot(1-p_2)\cdot r}\right).$$

The privacy level depends on p_1, p_2 and r. For $p_1 \neq 1$ and $p_2 \neq 1$, `RRTxFM` satisfies the definition of differential privacy. That is, r is not strictly required to guarantee differential privacy.

In Figure 5, we show ϵ_0 and ϵ_1 (absence and presence of an ID) in relation to p_2. The influence of r and p_1 are represented by the different lines. As for `RSTxFM`, r has no influence on ϵ_0. For high values of p_2, ϵ_0 increases quickly. We observe that for an increasing p_1 (i.e., truthful answers), ϵ_0 and ϵ_1 increase.

With rising p_1 the privacy level ϵ_1 becomes flatter. While r is not strictly required to gain differential privacy, it still has an influence on ϵ_1. The privacy level decreases with increasing p_2. Higher values of r have a positive effect on the privacy and make the curve's slope smaller, effectively decreasing ϵ_1 and thus ϵ.

5 Evaluation

In this section, we examine the accuracy of our approaches with respect to the privacy level ϵ. In particular, besides comparing the accuracy of `RSTxFM` and `RRTxFM`, we evaluate the cost of privacy. To this end, we implemented a simulation and generated synthetic datasets with different cardinalities. Each dataset consists of N unique random numbers, which serve the purpose of IDs. Since we know the true cardinalities, we can calculate the error of our cardinality estimations and directly compare the different approaches.

From PCSA it is known that it has a standard error of $0.78/\sqrt{m}$ [14]. A higher number of sketches consequentially results in a better accuracy. Since the number of sketches has no impact on the privacy level, though, we set $m = 64$ and the length of each sketch to 64 bit, large enough to count $\approx 7 \cdot 10^{19}$ elements. Please note that this is one of the benefits of building upon PCSA instead of Bloom filters, because we can set these parameters independently of N.

The perturbation and randomization in our approaches can lead to negative estimations. As this makes no sense, we set negative estimations to zero. In order

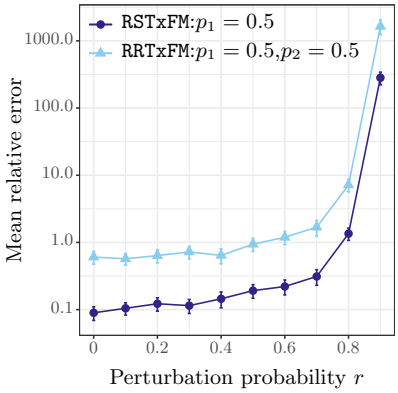

Fig. 6: Impact of r ($N = 10^4$).

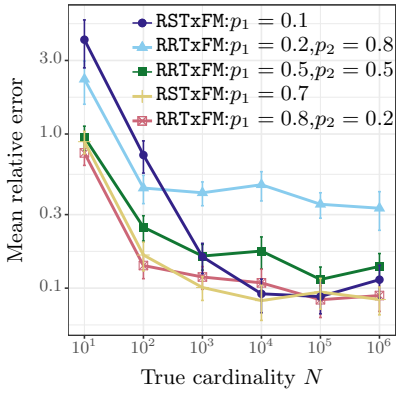

Fig. 7: Impact of p_1, p_2, and N.

to obtain statistically sound results, we repeat each experiment 50 times with varying random seeds. For all results, we show the arithmetic mean; error bars indicate 95% confidence intervals.

5.1 Impact of Parameters

We first investigate the accuracy of estimating the cardinality with varying perturbation probability r. In Figure 6, we show the relative error for `RSTxFM` and `RRTxFM`. For clarity, we set $p_1 = p_2 = 0.5$. For an increasing r, the relative error also increases. In line with the results of [23], the error remains at reasonable levels for $m = 64$ and $r < 0.4$. In the following experiments, we set $r = 0.2$, because we believe it provides a good trade-off between accuracy and privacy.

We also analyzed how the randomization parameters p_1 and p_2 and the cardinality size N influence the relative error. Probabilities were chosen to satisfy $\epsilon < 2$. Figure 7 generally indicates that the error decreases with higher cardinalities and stabilizes at some point. As expected, low cardinalities yield a high error. Also as expected, increasing p_1 decreases the error of both `RSTxFM` and `RRTxFM`. For `RRTxFM`, the probability p_2 (forced answers) also influences the accuracy. Increasing p_2 also increases the relative error. For high cardinalities, however, we observed that p_1 has the strongest impact on the loss of accuracy.

5.2 Cost of Privacy

Privacy comes at a cost. In order to quantify these costs, we compare our algorithms with vanilla PCSA (i.e., $r = 0$) and `PCSA(r)` with a perturbation probability $r = 0.2$. Please note that both, PCSA and `PCSA(r)`, do not satisfy the definition of differential privacy. Figure 8 shows that the relative error is less than 10%, even for small cardinalities. Figure 9 shows a trade-off between accuracy and privacy for `RSTxFM` and `RRTxFM`. As expected, more privacy guarantees (i.e., a smaller ϵ) yields a higher accuracy loss (particularly when compared to Figure 8). For a very small differential privacy level ($\epsilon = 0.23$), `RRTxFM` shows

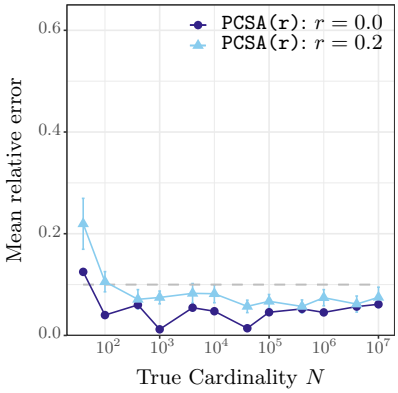

Fig. 8: Accuracy of PCSA.

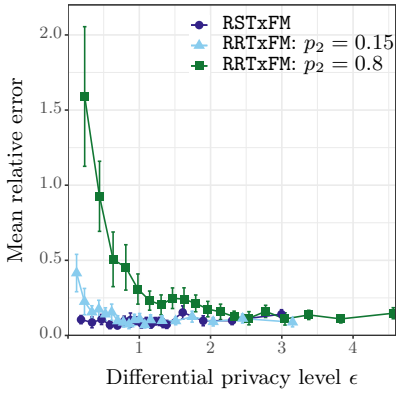

Fig. 9: Accuracy vs. Privacy ($N = 10^4$).

a high accuracy loss (error ≈ 1.6). `RSTxFM`, in contrast, is able to provide accurate results (error < 0.1) also for very small ϵ. Notably, the same privacy level does not lead to the same relative error. In particular, `RRTxFM` has a higher error for a higher probability p_2, even though the overall privacy guarantees are the same. This observation can also be made for higher cardinalities, where the error becomes even smaller. Lower cardinalities result in a higher error even with optimal parameters.

Table 1 summarizes the cost of privacy for appropriate parameters. As we mentioned above, stronger privacy comes at the cost of an increased loss of accuracy. However, the accuracy loss remains at a reasonable level for large cardinality sizes and an appropriate choice of parameters.

5.3 Discussion

In terms of the privacy level our approaches can be compared to RAPPOR [11] as it also uses RRT. The basic one-time RAPPOR guarantees differential privacy with $\epsilon \leq ln(3)$. With $\epsilon < 1$, we guarantee stronger privacy and an average error of less than 10%. According to [8], an error of less than 10% is classified as a precise cardinality estimator. To identify trends and reasons for employee turnover (as outlined in Example 1) this accuracy seems reasonable.

For the sake of clarity, we have envisioned our algorithms in a centralized setting so far only. That is, collecting data at a central point, which manages the sketches and performs the randomization. `RSTxFM` and `RRTxFM` however can also be used in a local mode and therefore provide *local differential privacy*. In this mode, each employee will manage the sketches and perform the described algorithm locally. The perturbed sketches will then be transmitted to the data collector. The perturbation r however will lead to a higher loss of accuracy when aggregating sketches. We therefore suggest to prefer `RRTxFM` for data collection and integration as it provides differential privacy even for $r = 0$.

When collecting data over time, the time series can leak information and eventually reveal the true value. In case of static already existing data, this is

Table 1: Cost of Privacy ($N = 10^4$; `RSTxFM`: $p_1 = 0.3$; `RRTxFM`: $p_1 = 0.4$, $p_2 = 0.15$).

		Relative Error			
Algorithm	r	Mean	Median	SD	Privacy
`PCSA(r)`	0.0	0.0476	0.0476	0.0	-
`PCSA(r)`	0.2	0.0820	0.0698	0.0624	only 1-bits
`RSTxFM`	0.2	0.0880	0.0658	0.0695	$\epsilon = 0.7885$
`RRTxFM`	0.2	0.0996	0.0659	0.0897	$\epsilon = 0.7777$

not relevant. However, it becomes relevant for employee satisfaction surveys, for example. RAPPOR provides protection against this type of information leakage by employing so-called memoization [11]. The memoization part "remembers" the result of RRT instead of recalculating it for a new query. This method is also applicable to `RRTxFM`. In the future, we will extend our evaluation and compare the results to various related approaches, including RAPPOR.

6 Conclusion

In this paper, we have shown that probabilistic counting can be used for differentially private statistics. We combined counting sketches with an additional randomization step to prevent personal data leakage. By comparing our developed algorithms, `RSTxFM` and `RRTxFM`, we exposed various parameter dependencies and found that the same privacy level does not necessarily result in the same accuracy. We however also showed that appropriate parameters can be found to gain privacy and accuracy.

In summary, our approaches provide strong differential privacy guarantees ($\epsilon < 1$) with a loss of accuracy below 10% and therefore balance the trade-off between privacy and accuracy.

References

1. Abbatiello, A., Agarwal, D., Bersin, J., et al.: The rise of the social enterprise, 2018 Deloitte Global Human Capital Trends. Deloitte (2018)
2. Alaggan, M., Cunche, M., Gambs, S.: Privacy-preserving Wi-Fi Analytics. Proceedings on Privacy Enhancing Technologies 2018(2), 4–26 (2018)
3. Alaggan, M., Gambs, S., Matwin, S., Tuhin, M.: Sanitization of call detail records via differentially-private Bloom filters. In: IFIP Annual Conference on Data and Applications Security and Privacy. pp. 223–230. Springer (2015)
4. Bianchi, G., Bracciale, L., Loreti, P.: Better Than Nothing Privacy with Bloom Filters: To What Extent? In: International Conference on Privacy in Statistical Databases. pp. 348–363. Springer (2012)
5. Bloom, B.H.: Space/time trade-offs in hash coding with allowable errors. Communications of the ACM 13(7), 422–426 (1970)
6. Boruch, R.F.: Assuring confidentiality of responses in social research: A note on strategies. The American Sociologist pp. 308–311 (1971)

7. Cormode, G.: Count-min sketch. Encyclopedia of Database Systems pp. 511–516 (2009)
8. Desfontaines, D., Lochbihler, A., Basin, D.A.: Cardinality Estimators do not Preserve Privacy. CoRR (2018)
9. Dwork, C., McSherry, F., Nissim, K., Smith, A.: Calibrating noise to sensitivity in private data analysis. In: Theory of cryptography conference. pp. 265–284. Springer (2006)
10. Dwork, C., Roth, A.: The Algorithmic Foundations of Differential Privacy. Foundations and Trends in Theoretical Computer Science 9(3-4), 211–407 (Aug 2014). https://doi.org/10.1561/0400000042
11. Erlingsson, Ú., Pihur, V., Korolova, A.: Rappor: Randomized aggregatable privacy-preserving ordinal response. In: Proceedings of the 2014 ACM SIGSAC conference on computer and communications security. pp. 1054–1067. ACM (2014)
12. Fan, L., Jin, H.: A Practical Framework for Privacy-Preserving Data Analytics. In: Proceedings of the 24th International Conference on World Wide Web. pp. 311–321 (2015). https://doi.org/10.1145/2736277.2741122
13. Fitz-Enz, J.: THE NEW HR ANALYTIC Predicting the Economic Value of Your Companys Human Capital Investments (2010)
14. Flajolet, P., Martin, G.N.: Probabilistic counting algorithms for data base applications. Journal of computer and system sciences 31(2), 182–209 (1985)
15. Kamp, M., Kopp, C., Mock, M., Boley, M., May, M.: Privacy-preserving mobility monitoring using sketches of stationary sensor readings. In: Joint European Conference on Machine Learning and Knowledge Discovery in Databases. pp. 370–386. Springer (2013)
16. Lieven, P., Scheuermann, B.: High-Speed Per-Flow Traffic Measurement with Probabilistic Multiplicity Counting. In: INFOCOM. pp. 1253–1261 (2010)
17. Lin, B., Wu, S., Tsou, Y., Huang, Y.: Ppdca: Privacy-preserving crowdsensing data collection and analysis with randomized response. In: 2018 IEEE Wireless Communications and Networking Conference (WCNC). pp. 1–6 (2018). https://doi.org/10.1109/WCNC.2018.8377050
18. Narayanan, A., Shmatikov, V.: Robust de-anonymization of large sparse datasets. In: Security and Privacy, 2008. SP 2008. IEEE Symposium on. pp. 111–125. IEEE (2008)
19. Qin, Z., Yang, Y., Yu, T., Khalil, I., Xiao, X., Ren, K.: Heavy hitter estimation over set-valued data with local differential privacy. In: Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security. pp. 192–203. ACM (2016)
20. Quoc, D.L., Beck, M., Bhatotia, P., Chen, R., Fetzer, C., Strufe, T.: Privapprox: privacy-preserving stream analytics. In: 2017 USENIX Annual Technical Conference. pp. 659–672 (2017)
21. Sparka, H., Tschorsch, F., Scheuermann, B.: P2KMV: A Privacy-preserving Counting Sketch for Efficient and Accurate Set Intersection Cardinality Estimations. Tech. Rep. 234 (2018)
22. Stanojevic, R., Nabeel, M., Yu, T.: Distributed Cardinality Estimation of Set Operations with Differential Privacy. In: 2017 IEEE Symposium on Privacy-Aware Computing (PAC). pp. 37–48 (Aug 2017). https://doi.org/10.1109/PAC.2017.43
23. Tschorsch, F., Scheuermann, B.: An algorithm for privacy-preserving distributed user statistics. Computer Networks 57(14), 2775–2787 (2013)
24. Warner, S.L.: Randomized response: A survey technique for eliminating evasive answer bias. Journal of the American Statistical Association 60(309), 63–69 (1965)