Libra: High-Utility Anonymization of Event Logs for Process Mining via Subsampling

Gamal Elkoumy
University of Tartu, Tartu, Estonia
gamal.elkoumy@ut.ee

Marlon Dumas
University of Tartu, Tartu, Estonia
marlon.dumas@ut.ee

***Abstract*—Process mining techniques enable analysts to identify and assess process improvement opportunities based on event logs. A common roadblock to process mining is that event logs may contain private information that cannot be used for analysis without consent. An approach to overcome this roadblock is to anonymize the event log so that no individual represented in the original log can be singled out based on the anonymized one. Differential privacy is an anonymization approach that provides this guarantee. A differentially private event log anonymization technique seeks to produce an anonymized log that is as similar as possible to the original one (high utility) while providing a required privacy guarantee. Existing event log anonymization techniques operate by injecting noise into the traces in the log (e.g., duplicating, perturbing, or filtering out some traces). Recent work on differential privacy has shown that a better privacy-utility tradeoff can be achieved by applying subsampling prior to noise injection. In other words, subsampling amplifies privacy. This paper proposes an event log anonymization approach called Libra that exploits this observation. Libra extracts multiple samples of traces from a log, independently injects noise, retains statistically relevant traces from each sample, and composes the samples to produce a differentially private log. An empirical evaluation shows that the proposed approach leads to a considerably higher utility for equivalent privacy guarantees relative to existing baselines.**

***Index Terms*—Process Mining, Event Log, Differential Privacy**

I. Introduction

Process Mining is a family of techniques to analyze event logs generated by enterprise information systems to help organizations identify opportunities to enhance their operational efficiency, compliance, and quality of service [7]. The primary input of a process mining technique is an event log. Each event represents the execution of an instance of an activity. An event contains a reference to a process instance (case identifier), a reference to an activity label, and at least one timestamp. An event may include other attributes, such as the resource (e.g., worker) who performed the activity. Table I presents an excerpt of a healthcare process event log.

Event logs may contain private information. For example, an adversary (or a data analyst) may use the sequence of activities performed in a case to single out patients. For example, patient 5 in Table I is the only case with the sequence <Registration, Antibiotics, Release>; given that Alice is the only patient who got antibiotics treatment in the log, the adversary can link case 5 to Alice. Also, the correlation between activities and their execution timestamp could be used to single out patients. For example, patient 1 in Table I is the only patient who registered in the morning on the 8th of April; since Bob is the only patient who was at the registration desk in the morning on that day, the adversary can link case 1 to Bob. Thus, given a released log, an adversary can perform a re-identification attack and single out an individual [14].

TABLE I
Example of an Event Log

Case ID	Activity	Timestamp	Other Attributes
1	Registration	4/8/2021 10:20	….....
	Triage	4/8/2021 10:50	….....
	Admission	4/8/2021 16:15	….....
2	Registration	4/8/2021 12:37	….....
	Admission	4/8/2021 14:37	….....
	Surgery	4/8/2021 15:07	….....
	Release	4/8/2021 20:31	….....
3	Registration	4/9/2021 13:30	….....
	Triage	4/9/2021 13:55	….....
	Admission	4/9/2021 20:55	….....
4	Registration	4/9/2021 15:00	….....
	Admission	4/9/2021 17:00	….....
	Surgery	4/9/2021 17:40	….....
	Release	4/9/2021 23:05	….....
5	Registration	4/9/2021 17:25	….....
	Antibiotics	4/9/2021 17:55	….....
	Release	4/10/2021 23:55	….....

Data regulations, such as GDPR[1], require organizations to put in place mechanisms to protect information about individuals when processing a dataset. One way to address this requirement is by using *Differential Privacy* (DP). DP stands out among other Privacy-Enhancing Techniques (PETs) due to its composability and privacy guarantees [10]. In a nutshell, DP is a guarantee to each individual that the ability for the analyst to infer private information about them will likely be the same whether their record(s) is/are part of the dataset or not. In other words, DP prevents singling out an individual after disclosing the data [6]. DP works by injecting noise into the data. This noise is quantified by a *privacy budget* parameter called ϵ, which captures the extent to which the presence or absence of the record(s) associated to an individual person in the original data, affect the disclosed (anonymized) data. The smaller the ϵ value, the stronger the privacy guarantee and the larger the injected noise. Thus, by controlling the privacy budget, we can ensure that the resulting anonymized dataset provides privacy guarantees, while still being useful for analysis. For example, in a healthcare event log, we can control the ϵ privacy budget to ensure that we can draw conclusions about bottlenecks and

Work funded by the European Research Council (PIX project)

[1] http://data.europa.eu/eli/reg/2016/679/oj

improvement opportunities from the anonymized log, while providing privacy guarantees to the patients whose clinical trajectories are represented by the traces in the log.

In this setting, this paper addresses the following problem: *Given an event log L, wherein each trace contains private information about an individual (e.g. a customer), and given a privacy budget ϵ, generate an anonymized event log L' that provides an ϵ-differential privacy guarantee to each individual represented in the log.*

Existing approaches for event log anonymization based on differential privacy [15], [17], [18], [20] inject noise to anonymize the frequency distribution of distinct sequences of activities in a log (a.k.a. the trace variants) and the event timestamps. This noise injection may introduce behavior not observed in the original log. The extent to which a DP-anonymization technique distorts the data is called the *utility loss*. The holy grail of anonymization techniques in general, and DP-anonymization techniques in particular, is to achieve a low level of re-identification risk with low utility loss.

Recent work in the field of DP has shown that the privacy guarantees of a differentially private mechanism can be amplified by applying it to a small random subsample of records [2]. This property is known as *privacy amplification*. The underpinning idea is that there is less utility loss overall since we inject less noise into the subsampled records. In this paper, we hypothesize that a DP approach based on subsampling can achieve lower utility loss for a given level of privacy guarantee relative to existing DP-anonymization techniques for event logs, which are based purely on noise injection.

The contribution of the paper is a DP-anonymization approach for event logs, namely Libra. Libra starts by filtering out trace variants that, if disclosed, would lead to privacy breaches. It then extracts multiple Poisson subsamples and applies a DP mechanism to anonymize each subsample. The resulting differentially private subsamples are combined to construct an anonymized log. Using the differential privacy composition theorem [11] and the privacy amplification results associated with *Renyi Differential Privacy* (RDP) [28], we estimate the amplified ϵ' privacy guarantee provided by the anonymized log. The paper reports on an empirical evaluation to assess the utility loss resulting from the anonymization procedure w.r.t the task of discovering the process map (Directly-Follows Graph) of an event log. The evaluation compares Libra against state-of-the-art approaches using eight real-life logs.

The paper is structured as follows. Sect. II introduces the background and related work. Sect. III presents the anonymization approach. Sect. IV presents an empirical evaluation. Finally, Sect. V concludes and discusses future work.

II. Background and Related Work

This section introduces differential privacy (DP) and formalizes its definition. We then overview the related work of privacy-preserving process mining (PPPM).

A. Differential Privacy (DP)

A dataset D consists of data points (entire cases in the event logs) that are drawn from a universe U. Two data sets D, and D' are called neighboring (adjacent) data sets if they differ in one data point at most. The two data sets D, D' can differ in: (1) the existence of a data point, which is called unbounded differential privacy [8]; (2) the value of a data point, which is called bounded differential privacy [9].

Definition 2.1 (Unbounded (ϵ, δ)-Differential Privacy) [8]): A mechanism M is (ϵ, δ) differentially private if every neighboring pair of datasets $D, D' \in U$ differing at most on one data point, and all $S \subseteq Range(M)$, the following inequality holds: $Pr[M(D) \in S] \leq e^{\epsilon} \cdot Pr[M(D') \in S] + \delta$.

Definition 2.2 (Bounded(ϵ, δ)-Differential Privacy) [9]): A mechanism M is (ϵ, δ) differentially private if every neighboring pair of datasets $D, D' \in U$ differing at most on the value of one data point in an attribute A, and all $S \subseteq Range(M)$, the following inequality holds: $Pr[M(D.A) \in S] \leq e^{\epsilon} \cdot Pr[M(D'.A) \in S] + \delta$.

The definitions guarantee that it is information-theoretically challenging to infer whether a data point belongs to either D or D' with certain confidence δ. ϵ and δ are the privacy parameters. ϵ controls the privacy loss, and intuitively, δ is the probability that the privacy loss would be greater than ϵ. The smaller the parameters are, the stronger the privacy is. In practice, a typical value for ϵ is 1, and for δ is $\frac{1}{N}$ where N is the number of data points in the dataset. The notion (ϵ, δ)-DP turns to be ϵ-DP when δ is 0.

A differentially private mechanism M_f injects a noise to a query result f. The sensitivity of f controls how much difference in the output depends on the input.

Definition 2.3 (Global Sensitivity): Let $f : D \rightarrow \mathbb{R}^d$.

- Global sensitivity w.r.t. the presence of an item is $\Delta f = \max_{D,D'} |f(D) - f(D')|$;
- Global sensitivity w.r.t. the value of an item in an attribute A is $\Delta^A f = \max_{D,D'} |f(D.A) - f(D'.A)|$;

where max is computed over every neighboring pair of datasets D, D' differing on the value of one item.

Given the dataset D, and the query f, a mechanism M_f returns a noisified result $f(D) + Y$, where Y is the generated noise. The injected noise is randomly drawn from a particular distribution. One commonly used distribution for the numeric queries is the Laplace distribution $Lap(b, \mu)$. The probability density function of the Laplace distribution is $\frac{1}{2b} exp(-\frac{|x-\mu|}{b})$, where μ is the mean, and b is the scale factor. With $Lap(b, 0)$, we can obtain ϵ-DP mechanism [10].

The classical design of DP mechanisms takes the parameters ϵ and δ as inputs and then carefully injects randomness to provide the privacy guarantee. This randomness can be achieved via sampling [15] or via changing the values of an attribute [17], [20]. Simultaneously, the DP mechanism tries to preserve the utility. However, this paradigm has been shifted recently since better privacy-utility tradeoffs can be achieved using a fine-grained analysis adapted to a specific mechanism [1], [2], [27].

Recent studies [2], [27] have shown that subsampling can be used in designing DP mechanisms that provide privacy amplification. Roughly speaking, if we apply an (ϵ, δ) differentially private mechanism to a γ-proportion random subsample of the dataset, the output satisfies $((O(\gamma\epsilon), \gamma\delta))$-DP [2]. This paper adopts Poisson subsampling of the cases in an event log to provide more optimal use of the privacy budget [28].

B. Privacy-Preserving Process Mining

One of the earliest PPPM approaches was proposed by Manndart et al. [20], which provides differential privacy for process mining. They anonymize the query "frequencies of directly-follows relations" and the query "the frequencies of trace variants". Furthermore, PRIPEL [17] adopts differential privacy to anonymize event logs. It embraces sequence enrichment for injecting noise to timestamps and other attributes of the event log. The above approaches accept three parameters: ϵ, k, and N. k is the cut-off frequency to filter out variants that occur less than k, and N is the length of the longest trace in the anonymized logs. The output of the above approaches may contain variants that were never observed in the event log. Besides, some of the newly injected variants are impossible to occur for the anonymized process [18].

SaCoFa [18] is another differentially private mechanism that achieves lower utility loss than the above approaches by using semantic constraints. This approach adopts DP to replace prefixes common in multiple cases with perturbed ones, given that the distance between the prefixes and the original one does not exceed a certain distance. However, this approach suffers from false negatives (traces that do not happen in the event log) and false positives (traces that occurred in the original log but do not appear in the anonymized log).

Elkoumy et al. [15] propose an approach that provides differentially private anonymization of event logs. They provide a privacy guarantee that the probability of singling out an individual in the anonymized log does not exceed a given threshold. They provide that guarantee while preserving the same trace variant as the input log. They adopt oversampling as their noise injection mechanism. However, this approach results in high noise injection over the directly-follow frequencies between activities.

Other studies adopt group-based models, as in [3], [16], [23], [24]. These approaches suppress entire cases or individual events (activity instances). When a technique suppresses an activity instance in a case, it may introduce behavior (e.g., a directly-follows relation) that does not exist in reality. When this suppression occurs at scale, the anonymized log contains a high proportion of behavior that does not happen in the original log. In a patient treatment log used in the evaluation reported in this paper, existing k-anonymization techniques for process mining may lead to the suppression of 87% of the activities [24].

Other studies of PPPM fall outside the scope of this paper, as they do not provide a mechanism to anonymize event logs. For example, privacy-preserving process performance indicators have been proposed in [19]. A risk quantification method has been presented in [26]. Privacy-preserving continuous event data processing has been studied in [22]. Other studies provided secure processing for distributed event logs in inter-organizational settings [12], [13].

III. Approach

We seek to anonymize an event log in such a way that an attacker cannot single out an individual (e.g., a patient) in the anonymized event log. Accordingly, Libra applies differential privacy to the event log. To provide better usage of a given privacy budget ϵ, Libra relies on privacy amplification by subsampling [2], [28].

More concretely, given an event log and a privacy budget ϵ, Libra produces a differentially-private event log in 5 steps as outlined in Fig.1. Some trace variants may be rare; keeping them can lead to singling out individuals. Accordingly, the first step clips trace variants below a clipping threshold C, estimated by the given privacy budget ϵ. The second step constructs a Poisson subsample of the event log. The third step is to anonymize subsamples. Some of the anonymized cases may affect the utility. Therefore, Libra performs statistical post-processing of the anonymized subsamples to select the relevant traces. Libra repeats the above process to generate multiple subsamples while tracking the privacy budget by noise screening, and it composes the privacy budgets of the subsamples using Renyi Differential Privacy (RDP). Lastly, Libra combines the generated subsamples to construct the anonymized event log. In the following, we introduce the attack model and explain each step of Libra in turn. Furthermore, an algorithm is presented to formalize Libra's steps.

A. Attack Model

We consider a setting where an event log publisher shares the log with an analyst who should not be able to link a case to an individual (e.g., a customer). An event log captures the activities of process execution, as illustrated in Table I. Each log record is an event with an activity label, an identifier of the process instance (Case ID), and a timestamp. An event may have other attributes, such as resources. Libra anonymizes event logs from the control flow perspective, representing the traces of individuals under study, and from the time perspective. Thus, we assume the input event log has only the three attributes: Case ID, activity label, and timestamp. A trace represents the sequence of events executed for a case.

Definition 3.1 (Event Log, Event, Trace): An event log $L = \{e_1, e_2, ..., e_n\}$ is a set of events $e = (i, a, ts)$, each capturing an activity execution a for a process instance (case) i, at timestamp ts. A trace $t = \langle e_1, e_2, ..., e_m \rangle$ of L is a chronologically ordered group of events $e_k \in L$ that share the same case identifier. Herein, we write $Traces(L)$ to refer to the set of traces of a log, and z to refer to the number of traces, i.e. $z = |Traces(L)|$.

We wish to protect the privacy of individuals (customers in this setting) from an analyst (the attacker in this setting) who seeks to determine if an individual participated in the log or not. Based on that, the attacker can link a case to an individual. An attacker can determine the existence of an individual in the released event log by using either the sequence of activities of a trace (control flow) or the execution time of a particular activity. Specifically, we seek to anonymize log L in such a way to prevent the attacker from achieving the following goals:

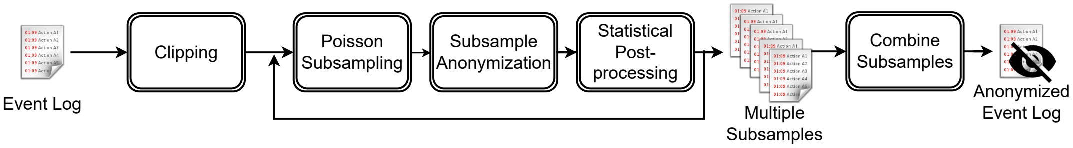

Fig. 1. Overview of the Libra's approach

- h_1: Distinguishing whether an individual has participated in the log or not through their execution control flow.
- h_2: Determining the execution timestamp of an activity.

We assume that the set of activity labels is public. Also, we assume that each individual (customer) participates only in one trace and that the execution of traces is independent. In case of an individual participates in more than one trace execution, the ϵ parameter shall be divided by the maximum number of traces an individual can participate in. That estimates the decreased ϵ-differential privacy guarantee. Following, we introduce the concept of a differentially-private event log that prevents an attacker from achieving the goals h_1 and h_2.

Definition 3.2 (Differentially-Private Event Log): Given an event log L as defined in Def. 3.1, an anonymized log L′ is said to be ϵ-differentially private if: (1) it provides ϵ-differential privacy over the control flow of traces; (2) the timestamp attribute is ϵ-differentially private.

B. Clipping Rare Traces

As explained above, an individual could be singled out through their control flow if they have a unique sequence of activities or trace. The set of the unique traces in the event log is called trace variants.

Definition 3.3 (Trace Variants): A trace variant of an event log L is a sequence of activities $\langle a_1, a_2, ..., a_k \rangle$ such that there is a trace $\langle e_1, e_2, ..., e_k \rangle$ of L such that $\forall j \in [1..k] : \ e_j = a_j$.

It is possible to single out individuals using their trace variants, even if the variant is not unique, but exists a few times in the log. We call such a trace variant a *Rare trace variant* [5]. Observing a rare trace variant is problematic because it is executed for a group of few individuals. Observing such a trace may increase the attacker's confidence about this group of individuals. Chaudhuri et al. [5] define rare values (rare trace variants) in a data sample w.r.t (ϵ, δ)-DP guarantee.

Lemma 3.1 (Rare Trace Variants [5]): A sample that provides (ϵ, δ)-differential privacy guarantee should not contain rare trace variants that happen less than $1/\epsilon \ log(\frac{2k}{\delta})$, where k is the number of trace variants in the log.

Since Libra provides privacy amplification via subsampling, any log sample that provides (ϵ, δ)-DP guarantee should not contain rare trace variants [5]. If a rare trace variant exists in the released log, it may lead to a privacy breach of $O(\frac{i\epsilon}{log(2k/\delta)})$, where i is the number of rare traces in the log [5]. Libra clips trace variants that happen less than $C = 1/\epsilon \ log(\frac{2k}{\delta})$. It does so by filtering out all the trace instances of the trace variants that happen $< C$.

Libra does not apply when the log is a collection of unique traces (i.e., each case is of a unique trace variant). The reason is that if every trace in the log is unique, then a sample of any size, even one, violates the privacy guarantees.

C. Event Log Subsampling

Our goal is to prevent an attacker from determining the existence of an individual in the anonymized log. To this end, we apply privacy amplification by subsampling [2], [28]. Specifically, we use Poisson subsampling as in [28].

Definition 3.4 (Poisson Subsample (after [28])): Given an event log L, a Poisson subsample of $L - PoissonSubsample(L)$ – is a subset of $Traces(L)$ obtained by randomly sampling traces, independently, from a Bernoulli distribution with parameter $\gamma = \frac{m}{z}$, where m is the sample size and $z = |Traces(L)|$.

The above procedure is equivalent to performing subsampling without replacement with sampling rate $\sim Binomial(\gamma, z)$. A binomial distribution converges to a Poisson distribution with parameter λ at the limit of $k \rightarrow \infty, \gamma \rightarrow 0$ while $\gamma z \rightarrow \lambda$.

Libra applies Poisson subsampling to the input event log and generates multiple subsamples. Each subsample contains entire traces rather than subtraces. That prevents the subsampling step from introducing trace variants that do not exist in the log. The next step explains the anonymization of each subsample.

D. Subsamples Anonymization

At this step, Libra provides an ϵ-DP guarantee for each subsample. Privacy amplification is achieved by applying a DP mechanism to a subsample of the dataset [2], [27].

Lemma 3.2 (Privacy Amplification [2], [28]): If a mechanism M is (ϵ, δ)-DP, then a subsampled mechanism M ∘ PoissonSubsample provides (ϵ', δ')-DP with $\epsilon' = log(1 + \gamma(e^{\epsilon} - 1))$ and $\delta' = \gamma\delta$.

Following, we explain how Libra applies differential privacy for each subsample (M ∘ PoissonSubsample). The goal of subsamples anonymization is to prevent attacks h_1 and h_2.

Definition 3.5 (Differentially-Private Subsample): Given a Poisson subsample S as defined in Def. 3.4, the output of a mechanism $S' = M(S)$ is said to be ϵ-differentially private if: (1) it provides ϵ-differential privacy over the set of trace variants; (2) the timestamp is ϵ-differentially private.

To anonymize the subsamples, we could use any of the DP mechanisms in the literature [15], [17], [18], [20]. We use the mechanism presented in [15] because it does not introduce trace variants that do not exist in the original log. It employs a guessing advantage to estimate the differential privacy parameter ϵ. Then, it injects the noise over a log representation called DAFSA. The noise injection for trace variants is done by oversampling traces. Libra adopt the approach in [15] to sample traces (adding or deleting traces) instead of oversampling. Also, we provide it with an ϵ value rather than a guessing advantage threshold. In the evaluation section, we measure the impact of privacy amplification on utility loss, and we compare both Libra and the one in [15].

To protect against linking a case to an individual via the cycle time of an activity (h_2), we inject noise drawn from a Laplace distribution quantified by a given privacy budget ϵ. Therefore, we anonymize two components of the timestamps: (1) case start time, which is the timestamp of the first event of the case; (2) the execution timestamp of every other event after that. In order to inject noise to the start time, we introduce the relative start time of the case, which is the difference between the case start time and the first start time in the log. We inject random noise quantified by ϵ to the relative start time. This noise may be positive or negative – in the latter scenario, the order of the start times of the traces may be altered.

On the other hand, we aim to protect the cycle time of each event, which is the difference between its execution timestamp and the execution timestamp of its successor event. We inject randomly generated noise quantified by ϵ to the cycle time. Both the cycle time and the generated noise are measured in minutes. After injecting noise as mentioned above, we transform the anonymized relative start time and cycle time to timestamps again via addition. At the end of this step, we have a differentially private subsample of the log (M $\circ$ PoissonSubsample).

E. Statistical Post-processing of SubSamples

Anonymization perturbs the utility of event logs. Libra selects statistically significant traces out of the anonymized log to provide higher utility of the differentially private event log. This selection process is a post-processing step to the anonymized subsamples. The result of a post-processing step of differentially-private subsamples of a log provides the same differential privacy guarantees [10].

Proposition 1 (Differential Privacy Under Post-processing [10]): A post-processing procedure P of a subsampled log L′ provides (ϵ, δ)-DP guarantees, if and only if L′ is (ϵ, δ)-differentially private.

The proof of Prop. 1 is in [10] (c.f. Proposition 2.1).

Given the differentially private Poisson subsample of the log, the post-processing of the selected subsample provides (ϵ, δ)-DP guarantees. Libra uses statistical post-processing of the subsamples to pick the most relevant traces and reduce the utility loss. These relevant traces are assessed by a trace abstraction function $\psi : 2^{\omega} \rightarrow \chi$, where χ is the domain of information extracted from a trace [4]. This information can be related to the sequence of activities in a trace and the frequencies of activities. Bauer et al. [4] provide a log sampling mechanism that adopts a series of binomial experiments and picks traces that provide new information while being discovery sufficient with probability ρ. Libra adopts the work in [4] to pick the relevant traces out of the differentially private subsamples. A trace τ provides new information if its abstraction is far from the union of the abstraction, jointly derived from the subsamples. That should happen within a distance ω of the union abstraction. Thus, we consider the predicate [4]:

$$\pi^{\omega}(S'_e, \tau) \leftrightarrow d\big(\psi(\tau), \bigcup_{\tau' \in S'_e} \psi(\tau')\big) > \omega \ , \tag{1}$$

where π is the picked sample, S'_e is the differentially private subsample, and ψ is the used abstraction. A subsample is discovery sufficient w.r.t. an abstraction ψ, a distance parameter ω, and probability ρ as follows:

Definition 3.6 (Discovery Sufficiency [4]): A DP-subsample $S'_e \subset S'$ is (ρ, ω, ψ)-discovery sufficient, if there is a newly picked trace $\tau : \tau \in (S' \setminus S'_e)$ that: $p_{\pi}(S'_e, \tau) = P(\pi(S'_e, \tau) = 1 < \omega$, where p is a probability measure.

At the end of this step, Libra has (ϵ, δ)-differentially private subsamples of the log that have statistically relevant traces.

F. Combining Subsamples

In order to construct an (ϵ, δ)-differentially private event log with the number of traces as close as possible to the number of traces in the original log, z, Libra repeats the differentially private subsampling operation for a number of times equals $\eta = \gamma z$. The repetitive access of the log via DP-subsampling is identical to the composition of the multiple differentially-private mechanisms (one mechanism for each DP subsample).

Theorem 3.3 (Differential Privacy Composition [10]): Let M_1 and M_2 be ϵ_1, ϵ_2-differentially-private mechanisms. Then, the combination of the mechanisms $M_{1,2}(L) = (M_1(L), M_2(L))$ is $\epsilon_1 + \epsilon_2$-differentially private.
The proof of Theorem 3.3 is in [10] (c.f. Theorem 3.14).

Indeed, the privacy parameters ϵ, δ degrade due to the composition. This degradation is compensated by the privacy amplification. One way to estimate ϵ, δ values after composition is using Renyi Differential Privacy (RDP) [21].

Definition 3.7 (Renyi Differential Privacy [21]): A mechanism M is said to be (α, ϵ)-RDP with order $\alpha \in (1, \infty)$ if for every neighboring datasets D,D′:

$$D_{\alpha}(M(D)||M(D')) = \frac{1}{\alpha - 1} log\Big(E_{\theta \ M(D')}[(\frac{P_{M(D)}(\theta)}{P_{M(D')}(\theta)})^{\alpha}]\Big) \leq \epsilon.$$

α is the Renyi divergence order between the two datasets D, D'. Mironov [21] provides an estimation of the equivalent ϵ of the RDP as a function of α. The estimated ϵ for the Laplace distribution equals:

$$\epsilon_{Laplace}(\alpha) = \frac{1}{\alpha - 1} \ log\Big(\frac{\alpha}{2\alpha - 1} e^{\frac{\alpha - 1}{b}} + \frac{\alpha - 1}{2\alpha - 1} e^{\frac{-\alpha}{b}}\Big) \tag{2}$$

The ϵ estimated in Eq. 2 can be passed to the Laplace mechanism to draw the noise and anonymize the subsamples. Zhu et al. [28] estimate the amplified ϵ' after Poisson subsampling and composition using RDP as:

$$\begin{aligned} \epsilon'_{M \circ PoissonSubsample}(\alpha) \leq \frac{1}{\alpha - 1} log\Big((1 - \gamma)^{\alpha - 1}(\alpha\gamma - \alpha + 1) \\ \binom{\alpha}{2} \gamma^2 (1 - \gamma)^{\alpha - 2} e^{\epsilon(2)} \\ +3 \sum_{j=3}^{\alpha} \binom{\alpha}{j} (1 - \gamma)^{\alpha - j} \gamma^j e^{(j-1)\epsilon(j)}\Big) \end{aligned} \tag{3}$$

G. Libra Anonymization Algorithm

We exploit the above observation as formalized in Alg. 1. Libra takes as input an event log L, an order of RDP α, a differential privacy parameter δ, and a sampling ratio γ. The algorithm estimates the ϵ used to draw noise from the Laplace distribution using Eq. 2. Then, it sets k to equal the count of trace variants in L. The algorithm estimates the clipping threshold C as explained in Lemma 3.1. The algorithm uses the estimated C to filter out trace variants that occur less than C from the log L. We set z to equal the number of cases in the filtered log $\hat{L}$. Then, the algorithm estimates η, the number of subsamples needed to construct an event log with as many cases as the filtered log $\hat{L}$. After that, the algorithm performs Poisson sampling over $\hat{L}$. Given the Poisson subsample S, the algorithm generates an anonymized sample S' as explained in Sec. III-D. Then, it picks the statistically relevant traces from the anonymized sample S' as explained in Sec. III-E. The algorithm generates η subsamples and combines them to generate L'. Also, the algorithm reports the amplified ϵ' after the composition of privacy budgets of subsamples using Eq. 3.

Algorithm 1: The Libra Anonymization Algorithm

```
    Input: $L$: Event Log,
$\alpha$: order of Renyi Differential Privacy,
$\delta$: Differential Privacy parameter,
$\gamma$: Sampling Ratio.
    Output: $L'$: $\epsilon'$-Differentially Private Event Log,
$\epsilon'$: the amplified differential privacy guarantee.
$\epsilon \leftarrow$ EstimateEpsilon($\alpha$);            ▷ Estimate $\epsilon_{Laplace}$ using Eq. 2.
$k \leftarrow$ CountVariants($L$);            ▷ Set $k$ to the count of trace variants in $L$.
$C \leftarrow (1/\epsilon)\ log(2k/\delta)$;  ▷ Estimate clipping threshold based on Lemma 3.1.
$\hat{L} \leftarrow$ Filter(L, C);            ▷ Filter out trace variants with frequency below C.
$z \leftarrow$ CountCases($\hat{L}$);            ▷ Set $z$ to the count of cases in $\hat{L}$.
$\eta \leftarrow \gamma z$;            ▷ Calculate the count of subsamples.
$L' \leftarrow 0$;
while $i < \eta$ do
    S= PoissonSubsample($\hat{L}, \gamma$);  ▷ Perform Poisson subsampling as defined
      in Def. 3.4.
    $S' \leftarrow$ Anonymize($S, \epsilon$);     ▷ Anonymize the subsample as explained in
      Sec. III-D.
    $S'_e \leftarrow$; StatisticalPost-processing($S'$);            ▷ Perform statistical
      post-processing to select relevant cases as explained in Sec. III-E.
    $L' \leftarrow L' \cup S'_e$;
    i++ ;
end
$\epsilon' \leftarrow$ EstimateComposition($\alpha, \epsilon, \gamma$);            ▷ Estimate $\epsilon'$ using Eq. 3.
return $L', \epsilon'$
```

Alg. 1 employs a differentially private mechanism to anonymize every Poisson subsample and performs post-processing of subsamples to pick the relevant traces. Since the post-processing depends only on the anonymized samples, the output of the post-processing (DP-subsamples) is differentially private, c.f., Prop. 1. After that, the algorithm uses differentially private composition to combine the differentially-private subsamples to construct the anonymized event log. Thus, the algorithm's output is differentially private.

IV. Evaluation

To address the research problem stated in Sect. I, Libra injects noise to provide (ϵ, δ)-DP guarantees in two ways: (1) by filtering out rare trace variants and sampling traces in the log; (2) by introducing time shifts to the event timestamps. Filtering and noise injection affect the utility of the anonymized logs. To measure this utility loss, we compare the anonymized event logs against the original logs using a distance measure. Also, we compare the performance (execution time) of Libra against state-of-the-art baselines.

The evaluation reported below is driven by the following research questions:

RQ1. Does Libra outperform the state-of-the-art baselines in terms of the output utility?

RQ2. What is the difference between Libra and the state-of-the-art in terms of computational efficiency?

A. Evaluation Measures

Given an event log, mainstream process mining tools generally produce a Directly-Follows Graph (DFG) to capture the behavior of the process for the purpose of process discovery and analysis. In addition to being used as a visualization technique on its own, DFGs are the starting point for several algorithms to discover process models from event logs. Accordingly, and in line with prior work, we measure utility loss over the DFG of an event log. A DFG can be seen as a function that maps each pair of consecutive activities in the log (i.e. each directly-follows relation) to its frequency. This function can be seen as a frequency distribution. Accordingly, to measure utility loss over DFGs, we use the *Earth Movers' Distance*(EMD) [25]. The EMD distance between two distributions u and v is the minimum cost of transforming u into v. The cost is the weight of the distribution to be moved, multiplied by the distance it moves. Formally:

$$EMD(u, v) = \inf_{\pi \in \Gamma(u,v)} \int_{\mathbb{R} \times \mathbb{R}} |x - y| d\pi(x, y), \qquad (4)$$

where $\Gamma(u, v)$ is the set of distributions on $\mathbb{R} \times \mathbb{R}$ whose marginals are u and v.

In order to measure the computational efficiency of Libra, we measure the wall-to-wall run time experiment. We measure the time between submitting an event log to the approach and receiving its anonymized version.

B. Datasets

In our evaluation, we rely on the publicly available event logs of 4TU Centre for Research Data[2] as of February 2022. The selected logs and their characteristics are presented in the supplementary material[3].

C. Experiment Setup

We implement the proposed model as part of Libra[4] prototype. We run the experiment on a single machine with AMD Opteron(TM) Proc 6276 and 32 GB RAM. We time out any experiment at 24 hours. Also, in our experiment, we consider only the end timestamp to calculate the relative time of an event for simplicity. The approach still provides DP

[2] https://data.4tu.nl/
[3] https://doi.org/10.5281/zenodo.6376761
[4] https://github.com/Elkoumy/Libra

guarantees for logs with start and end timestamps. We inject noise to the relative start time measured in days.

We fix the parameters b=2, γ=0.05, and $\delta = 10^{-4}$, and we evaluate the approach for different values of α. An empirical evaluation between γ, α, and ϵ under Poisson subsampling, is presented in [28].

We compare our approach against the state-of-the-art. The studies that address the same problem are [15], [17], [18], [20], [23], [24]. We exclude the work in [23], [24] from our experiment because the interpretation of the k-anonymity parameters and DP are different. The studies in [15], [17], [18], [20] provide DP guarantees. The work in [20] anonymizes two types of queries, but the output is not an event log. PRIPEL [17] adopts [20] for trace variant anonymization. We compare Libra against [17]. SaCoFa [18] provides trace variant anonymization but does not consider timestamp anonymization. Therefore, we consider [18] only in EMD of frequency annotated DFG experiments. Both PRIPEL and SaCoFa take three input parameters namely, ϵ, k, and N. For the pruning parameter k, we run several experiments with different values of k (0.5%, 1%,5% of the cases), and we select the best result. For the maximum trace length N, we set N to the average trace length of the log. Amun [15] anonymizes both the trace variants and the timestamps. It accepts the guessing advantage probability δ, rather than ϵ. Thus, we set δ to the values (0.01, 0.075, 0.4) that give an average ϵ close to the ϵ considered in the experiments. We use the EMD to compare the anonymized log against the original one for the selected approaches.

D. Results

We measure the EMD between the DFG of the anonymized log and the DFG of the original log. We report the differences in terms of the frequency and time (measured in hours). Table II shows the experimental results using the EMD distance. α refers to the RDP parameter, and ϵ' refers to the equivalent DP parameter after the amplification and composition. The best result for every input α is in bold. Libra outperforms the state-of-the-art baselines in most of the logs, because the privacy amplification reduces the amount of injected noise, and hence achieves a lower utility loss.

Conversely, Libra has a lower frequency EMD than $Amun_o$ ([15]) because the latter always injects positive noise, which affects the utility. Libra outperforms both PRIPEL and SaCoFa over the frequency EMD because for a given ϵ'-DP guarantee, Libra needs an $\epsilon > \epsilon'$ due to privacy amplification. On the other hand, in event logs with many rare cases (Lemma 3.1) such as Sepsis and Urineweginfectie logs. Libra does not outperform the state-of-the-art because it adopts clipping to get rid of rare trace variants. Due to clipping, Libra sometimes return an empty log, e.g., for Sepsis log with $\alpha = 2$ and Hospital log with $\alpha = 2$. Also, Libra has a lower utility loss over the anonymized timestamps (measured by EMD time), which happens due to privacy amplification.

We evaluate the processing efficiency of Libra via a wall-to-wall run time experiment. Table III presents the experiments for $\alpha = 10$ ($\epsilon' = 0.37$). The time is measured in seconds. This experiment excludes SaCoFa as it provides only trace variant anonymization. The run time increases with the increase of the log size. The above experiments have been performed using a single thread to avoid adding other variables to the experiments.

We acknowledge that Libra is not suitable for event logs with many rare trace variants (unstructured event logs). In such a case the approach filters out most of the cases, and sometimes it returns an empty event log. Also, the above observations are based on a limited population (8 event logs). However, we selected the logs from a broader real-life event log population.

V. Conclusion and Future Work

This paper proposed a differentially private mechanism to anonymize event logs for process mining. While previous proposals in this field rely purely on noise injection, the approach proposed in this paper additionally employs subsampling to achieve stronger privacy guarantees with the same level of utility loss, or conversely, less utility loss for the same privacy guarantee (privacy amplification). The empirical evaluation shows that the privacy amplification effect leads to significant reductions of utility loss, particularly when it comes to anonymizing the frequency of distribution of case variants in a log (i.e. control flow anonymization) and to a lesser extent when it comes to anonymization of event timestamps.

A limitation of Libra is that it is not suitable for event logs with a high proportion of infrequent trace variants. In such use cases, Libra simply filters out most of the traces and may lead to empty outputs. A possible future research avenue is to address this limitation by applying summarization techniques to replace groups of similar infrequent trace variants with a single prototypical trace variant, which would then have a frequency equal to the sum of the frequencies of the trace variants in the group. A second limitation of Libra is that it only anonymizes three columns of an event log (case ID, activity label, and timestamp). Real-life log may contain additional event attributes, e.g. resources, cost, location. Another avenue for future work is to extend the privacy composition mechanism to include the anonymization of additional attributes.

References

[1] Abadi, M., Chu, A., Goodfellow, I.J., McMahan, H.B., Mironov, I., Talwar, K., Zhang, L.: Deep learning with differential privacy. In: CCS. pp. 308–318. ACM (2016)

[2] Balle, B., Barthe, G., Gaboardi, M.: Privacy amplification by subsampling: Tight analyses via couplings and divergences. In: NeurIPS. pp. 6280–6290 (2018)

[3] Batista, E., Solanas, A.: A uniformization-based approach to preserve individuals' privacy during process mining analyses. Peer-to-Peer Netw. Appl. **14**(3), 1500–1519 (2021)

[4] Bauer, M., Senderovich, A., Gal, A., Grunske, L., Weidlich, M.: How much event data is enough? A statistical framework for process discovery. In: CAiSE. Lecture Notes in Computer Science, vol. 10816, pp. 239–256. Springer (2018)

[5] Chaudhuri, K., Mishra, N.: When random sampling preserves privacy. In: CRYPTO. Lecture Notes in Computer Science, vol. 4117, pp. 198–213. Springer (2006)

[6] Cohen, A., Nissim, K.: Towards formalizing the gdpr's notion of singling out. Proc. Natl. Acad. Sci. **117**, 8344–8352 (2020)

[7] Dumas, M., La Rosa, M., Mendling, J., Reijers, H.A., et al.: Fundamentals of business process management, vol. 1. Springer (2013)

TABLE II

EARTH MOVERS' DISTANCE FOR THE OUTPUT OF DIFFERENT ANONYMIZATION APPROACHES. A "-" MEANS THAT THE APPROACH RAN OUT OF MEMORY OR TIMED OUT, AND A "N/A" MEANS THE ENGINE RETURNS AN EMPTY LOG.

Log	α	ϵ'	EMD Freq				EMD Time		
			$Amun_o$	$Libra$	$PRIPEL$	$SaCoFa$	$Amun_o$	$Libra$	$PRIPEL$
BPIC12	2	0.04	-	1036	-	**1007**	-	**19315**	-
	10	0.37	38442	**997**	-	1007	15068248	**19228**	-
	100	4.7	1293	**1005**	1230	1007	54328	**19224**	**19224**
BPIC13	2	0.04	18853	**3847**	4954	4947	9520973	**145272**	197564
	10	0.37	19573	**3586**	4952	4948	10552209	**132827**	197564
	100	4.7	4436	**3492**	4952	4952	1208926	**127752**	197564
BPIC14	2	0.04	-	**494**	-	544	-	**7088**	-
	10	0.37	-	**466**	-	544	-	**6970**	-
	100	4.7	1013	**464**	-	507	128218	**6944**	-
BPIC17	2	0.04	-	**1563**	-	2785	-	**48254**	-
	10	0.37	-	**1349**	-	2789	-	**41190**	-
	100	4.7	**939**	1333	-	2790	148867	**40864**	-
BPIC20	2	0.04	-	**91**	128	125	-	**15602**	24422
	10	0.37	342	**82**	128	127	629659	**13920**	24422
	100	4.7	**79**	**79**	128	127	58201	**13504**	24430
Hospital	2	0.04	-	N/A	-	**35**	-	N/A	-
	10	0.37	-	**35**	-	**35**	-	**2507**	-
	100	4.7	-	**35**	**35**	**35**	-	2507	**2469**
Sepsis	2	0.04	1120	N/A	**75**	117	214068	N/A	**4644**
	10	0.37	1061	123	**120**	121	205646	**6218**	**6218**
	100	4.7	234	123	**120**	**120**	21661	**6218**	**6218**
Unrine.	2	0.04	1021	120	298	**42**	2921865	**58576**	68245
	10	0.37	169	95	292	**56**	547254	**49005**	68234
	100	4.7	31	81	292	**64**	52284	**37381**	68234

TABLE III

EXECUTION TIME EXPERIMENT. THE TIME IS MEASURED IN SECONDS FOR AN $\epsilon' = 0.37$. A "-" MEANS THAT THE APPROACH RAN OUT OF MEMORY OR TIMED OUT.

dataset	$Libra$	$Amun_o$	$PRIPEL$
BPIC12	90	503	-
BPIC13	49	115	306
BPIC14	323	-	-
BPIC17	212	-	-
BPIC20	70	170	330
Hospital	90	-	-
Sepsis	10	80	24
Unrine.	9	6.4	2

[8] Dwork, C.: Differential privacy: A survey of results. In: Proc. TAMC. pp. 1–19. Springer (2008)

[9] Dwork, C., McSherry, F., Nissim, K., Smith, A.: Calibrating noise to sensitivity in private data analysis. In: TCC. pp. 265–284 (2006)

[10] Dwork, C., Roth, A., et al.: The algorithmic foundations of differential privacy. Found. Trends Theor. Comput. Sci. **9**(3-4), 211–407 (2014)

[11] Dwork, C., Rothblum, G.N., Vadhan, S.P.: Boosting and differential privacy. In: FOCS. pp. 51–60. IEEE Computer Society (2010)

[12] Elkoumy, G., Fahrenkrog-Petersen, S.A., Dumas, M., Laud, P., Pankova, A., Weidlich, M.: Secure multi-party computation for inter-organizational process mining. In: BPMDS/EMMSAD@CAiSE. Lecture Notes in Business Information Processing, vol. 387, pp. 166–181. Springer (2020)

[13] Elkoumy, G., Fahrenkrog-Petersen, S.A., Dumas, M., Laud, P., Pankova, A., Weidlich, M.: Shareprom: A tool for privacy-preserving inter-organizational process mining. In: BPM (PhD/Demos). CEUR Workshop Proceedings, vol. 2673, pp. 72–76. CEUR-WS.org (2020)

[14] Elkoumy, G., Fahrenkrog-Petersen, S.A., Sani, M.F., Koschmider, A., Mannhardt, F., von Voigt, S.N., Rafiei, M., von Waldthausen, L.: Privacy and confidentiality in process mining: Threats and research challenges. ACM Trans. Manag. Inf. Syst. **13**(1), 11:1–11:17 (2022)

[15] Elkoumy, G., Pankova, A., Dumas, M.: Mine me but don't single me out: Differentially private event logs for process mining. In: ICPM. pp. 80–87. IEEE (2021)

[16] Fahrenkrog-Petersen, S.A., van der Aa, H., Weidlich, M.: PRETSA: event log sanitization for privacy-aware process discovery. In: ICPM. pp. 1–8. IEEE (2019)

[17] Fahrenkrog-Petersen, S.A., van der Aa, H., Weidlich, M.: PRIPEL: privacy-preserving event log publishing including contextual information. In: BPM. pp. 111–128. Springer (2020)

[18] Fahrenkrog-Petersen, S.A., Kabierski, M., Rösel, F., van der Aa, H., Weidlich, M.: SaCoFa: semantics-aware control-flow anonymization for process mining. In: ICPM (2021)

[19] Kabierski, M., Fahrenkrog-Petersen, S.A., Weidlich, M.: Privacy-aware process performance indicators: Framework and release mechanisms. In: CAiSE. Lecture Notes in Computer Science, vol. 12751, pp. 19–36. Springer (2021)

[20] Mannhardt, F., Koschmider, A., Baracaldo, N., Weidlich, M., Michael, J.: Privacy-preserving process mining. BISE **61**, 595–614 (2019)

[21] Mironov, I.: Rényi differential privacy. In: CSF. pp. 263–275. IEEE Computer Society (2017)

[22] Rafiei, M., van der Aalst, W.M.P.: Privacy-preserving continuous event data publishing. In: BPM (Forum). Lecture Notes in Business Information Processing, vol. 427, pp. 178–194. Springer (2021)

[23] Rafiei, M., van der Aalst, W.M.: Group-based privacy preservation techniques for process mining. Data Knowl. Eng. p. 101908 (2021)

[24] Rafiei, M., Wagner, M., van der Aalst, W.M.: TLKC-Privacy model for process mining. In: RCIS. pp. 398–416. Springer (2020)

[25] Ramdas, A., Trillos, N.G., Cuturi, M.: On wasserstein two-sample testing and related families of nonparametric tests. Entropy **19**(2), 47 (2017)

[26] von Voigt, S.N., Fahrenkrog-Petersen, S.A., Janssen, D., Koschmider, A., Tschorsch, F., Mannhardt, F., Landsiedel, O., Weidlich, M.: Quantifying the re-identification risk of event logs for process mining - empiricial evaluation paper. In: CAiSE. Lecture Notes in Computer Science, vol. 12127, pp. 252–267. Springer (2020)

[27] Wang, Y., Balle, B., Kasiviswanathan, S.P.: Subsampled renyi differential privacy and analytical moments accountant. In: AISTATS. Proceedings of Machine Learning Research, vol. 89, pp. 1226–1235. PMLR (2019)

[28] Zhu, Y., Wang, Y.: Poission subsampled rényi differential privacy. In: ICML. Proceedings of Machine Learning Research, vol. 97, pp. 7634–7642. PMLR (2019)