TraVaS: Differentially Private Trace Variant Selection for Process Mining

Majid Rafiei(✉), Frederik Wangelik, and Wil M. P. van der Aalst

Chair of Process and Data Science, RWTH Aachen University, Aachen, Germany
majid.rafiei@pads.rwth-aachen.de

Abstract. In the area of industrial process mining, privacy-preserving event data publication is becoming increasingly relevant. Consequently, the trade-off between high data utility and quantifiable privacy poses new challenges. State-of-the-art research mainly focuses on differentially private trace variant construction based on prefix expansion methods. However, these algorithms face several practical limitations such as high computational complexity, introducing fake variants, removing frequent variants, and a bounded variant length. In this paper, we introduce a new approach for direct differentially private trace variant release which uses anonymized *partition selection* strategies to overcome the aforementioned restraints. Experimental results on real-life event data show that our algorithm outperforms state-of-the-art methods in terms of both plain data utility and result utility preservation.

Keywords: Process mining · Differential privacy · Event data

1 Introduction

In recent years, process mining and event data analysis have been successfully deployed in many industries. The main objectives are to learn process models from event logs for further behavioral inference (so-called *process discovery*), to extend existing models using event logs (so-called *model enhancement*), or to assess the alignment between a process model and an event log (so-called *conformance checking*) [2]. However, often the underlying event data are bound to personal identifiers or other private information. A prominent example is the process management of hospitals where the cases are patients being treated by staff. Without means of privacy protection, any adversary is able to extract sensitive information about individuals and their properties. Thus, privacy regulations, such as GDPR [1], typically restrict data storage and access which motivates the development of privacy preservation techniques.

The majority of state-of-the-art privacy preservation techniques are built on Differential Privacy (DP), which offers a noise-based privacy definition. This is due to its important features, such as providing mathematical privacy guarantees and security against *predicate-singling-out* attacks [3]. The goal of techniques

Funded under the Excellence Strategy of the Federal Government and the Länder. We also thank the Alexander von Humboldt Stiftung for supporting our research.

M. Montali et al. (Eds.): ICPM 2022 Workshops, LNBIP 468, pp. 114–126, 2023.
https://doi.org/10.1007/978-3-031-27815-0_9

Table 1. A simple event log from the healthcare context including trace variants and their frequencies.

Trace variant	Frequency
$\langle register, visit, blood\text{-}test, release \rangle$	10
$\langle register, blood\text{-}test, visit, release \rangle$	8
$\langle register, visit, release \rangle$	20
$\langle register, visit, blood\text{-}test, blood\text{-}test, release \rangle$	5

based on DP is to hide the participation of an individual in the released output by injecting noise. The amount of noise is mainly determined by the privacy parameters, ϵ and δ, and the sensitivity of the underlying data. State-of-the-art research targeting (ϵ, δ)-DP methods in process mining focuses on releasing raw privatized activity sequences performed for cases, i.e., *trace variants*. Table 1 shows a sample of such event data in the healthcare context, where each trace variant belongs to a case, i.e., a patient, and one case cannot have more than one trace variant. This format describes the *control-flow* of event logs that is basis for the main process mining activities. The trace variant of a case is considered sensitive information because it contains the complete sequence of activities performed for the case that can be exploited to conclude private information, e.g., patient diseases in the healthcare context.

To achieve differential privacy for trace variants, the state-of-the-art approach [12] inserts noise drawn from a *Laplacian distribution* into the variant distribution obtained from an event log. This approach has several drawbacks including: (1) *introducing fake variants*, (2) *removing frequent true variants*, and (3) *limited length for generated trace variants*. A recent work called *SaCoFa* [9], attempts to mitigate drawbacks (1) and (2) by gaining knowledge regarding the underlying process semantics from original event data. However, the privacy quantification of all extra queries to gain knowledge regarding the underlying semantics is not discussed. Moreover, the third drawback still remains since this work, similar to [12], employs a *prefix-based* approach. The prefix-based approaches need to generate all possible unique variants based on a set of activities to provide differential privacy for the original distribution of variants. Since the set of possible trace variants that can be generated given a unique set of activities is infinite, the prefix-based techniques need to bound the length of generated sequences. Also, to limit the search space these approaches typically include a pruning parameter to exclude less frequent prefixes.

We introduce an (ϵ, δ)-DP approach for releasing the distribution of trace variants that focuses on the aforementioned drawbacks. In contrast to the prefix-based approaches, the underlying algorithm is based on (ϵ, δ)-DP for *partition selection* that allows for a direct publication of arbitrarily long sequences [4]. Employing differentially private partition selection techniques, the actual frequencies of all trace variants can directly be queried without guessing (generating) trace variants. Internally, random noise drawn from a specific geometric distribution is injected into the corresponding frequencies, and all variants whose privatized frequencies fall beyond a threshold are removed. Hence, no fake trace variants are introduced, and only some infrequent variants may disappear from the output. Moreover, no tedious fine-tuning has to be conducted and no computationally expensive search needs to be included. In Sect. 5, we introduce different metrics to evaluate the *data* and *result* utility preservation of our approach. We also run our experiments for the state-of-the-art prefix-based methods and show superior data and result utilities compared to these methods.

The remainder of this paper is structured as follows. In Sect. 2, we provide a summary of related work. Preliminaries and notations are provided in Sect. 3. Section 4 introduces the theoretical background of differentially private *partition selection*, and describes our *TraVaS* algorithm. In Sect. 5, the experimental results based on real-life event logs are shown. Section 6 concludes the paper.

2 Related Work

The research area of privacy and confidentiality in process mining is recently growing in importance. Several techniques have been proposed to address the privacy and confidentiality issues. In this paper, our focus is on the so-called *noise-based* techniques that are based on the notion of *differential privacy*. In [12], the authors apply an (ϵ, δ)-DP mechanism to event logs to privatize *directly-follows relations* and trace variants. The underlying principle uses a combination of an (ϵ, δ)-DP noise generator and an iterative query engine that allows an anonymized publication of trace variants with an upper bound for their length. *SaCoFa* [9] is the most recent extension of the aforementioned (ϵ, δ)-DP mechanism that attempts to optimize the query structures with the help of underlying semantics. Another extension of [12] is the *PRIPEL* approach, where more event attributes can be secured using the so-called *sequence enrichment* [8].

Whereas most of the aforementioned ideas target raw event logs, in [7], the focus is on *directly-follows graphs*. During the edge generation, connections are randomized using (ϵ, δ)-DP mechanisms to balance utility preservation and privacy risks. As the main benchmark model for our work, we choose the technique by Mannhardt et al. [12] since it focuses on trace variants and is the basis of most of the other techniques. Moreover, its privacy guarantees are directly proven by (ϵ, δ)-DP mechanisms, i.e., no extra privacy analysis is required. Nevertheless, we also compare our results with SaCoFa as the most recent extension of the benchmark to demonstrate the superior performance of our approach.

3 Preliminaries

In this section, we introduce the necessary mathematical concepts and definitions utilized throughout the remainder of the paper. Let A be a set. $B(A)$ is the set of all multisets over A. A multiset A can be represented as a set of tuples $\{(a, A(a)) | a \in A\}$ where $A(a)$ is the frequency of $a \in A$. Given A and B as two multisets, $A \uplus B$ is the sum over multisets, e.g., $[a^2, b^3] \uplus [b^2, c^2] = [a^2, b^5, c^2]$. We define a finite sequence over A of length n as $\sigma = \langle a_1, a_2, \ldots, a_n \rangle$ where $\sigma(i) = a_i \in A$ for all $i \in \{1, 2, \ldots, n\}$. The set of all finite sequences over A is denoted with A^*.

3.1 Event Data

The data used by *process mining* techniques are typically collections of unique events that are recorded per activity execution and characterized by their attributes. We denote $\mathcal{E}$ as the universe of events. Then, a *trace* σ, which is a single process execution, is represented as a sequence of events $\sigma = \langle e_1, e_2, ..., e_n \rangle \in \mathcal{E}^*$ belonging to the same case and having a fixed ordering based on timestamps. Note that events are unique and cannot appear in more than one trace. Moreover, each case (individual) contributes to only one trace. An event log L can be represented as a set of traces $L \subseteq \mathcal{E}^*$. Our work focuses on the control-flow aspect of an event log that only considers the activity attribute of events in traces. We define a simple event log based on activity sequences, so-called *trace variants*.

Definition 1 (Trace Variant). *Let $\mathcal{A}$ be the universe of activities. A trace variant $\sigma = \langle a_1, a_2, ..., a_n \rangle \in \mathcal{A}^*$ is a sequence of activities performed for a case.*

Definition 2 (Simple Event Log). *A simple event log L is defined as a multiset of trace variants $L \in B(\mathcal{A}^*)$. $\mathcal{L}$ denotes the universe of simple event logs.*

3.2 Differential Privacy

In the following, we introduce the necessary concepts of (ϵ, δ)-DP for our research. The main idea of DP is to inject noise into the original data in such a way that an observer who sees the randomized output cannot tell if the information of a specific individual is included in the data [6]. Considering simple event logs, i.e., the distribution of trace variants, as our sensitive event data, differential privacy can formally be defined as Definition 3.

Definition 3 ((ϵ,δ)-DP for Event Logs). *Let L_1 and L_2 be two neighbouring event logs that differ only in a single entry, e.g., $L_2 = L_1 \uplus [\sigma]$ for any $\sigma \in \mathcal{A}^*$. Also, let $\epsilon \in \mathbb{R}_{>0}$ and $\delta \in \mathbb{R}_{>0}$ be two privacy parameters. A randomized mechanism $\mathcal{M}_{\epsilon,\delta} : \mathcal{L} \rightarrow \mathcal{L}$ provides (ϵ, δ)-DP if for all $S \subseteq \mathcal{A}^* \times \mathbb{N}$: $Pr[\mathcal{M}_{\epsilon,\delta}(L_1) \in S] \leq e^{\epsilon} \times Pr[\mathcal{M}_{\epsilon,\delta}(L_2) \in S] + \delta$. Given $L \in \mathcal{L}$, $\mathcal{M}_{\epsilon,\delta}(L) \subseteq \{(\sigma, L'(\sigma)) \mid \sigma \in \mathcal{A}^* \wedge L'(\sigma) = L(\sigma) + x_\sigma\}$, with x_σ being realizations of i.i.d. random variables drawn from a probability distribution.*

In Definition 3, ϵ as the first privacy parameter specifies the probability ratio, and δ as the second privacy parameter allows for a linear violation. In the strict case of $\delta = 0$, $\mathcal{M}$ offers ϵ-DP. The randomness of respective mechanisms is typically ensured by the noise drawn from a probability distribution that perturbs original variant-frequency tuples and results in non-deterministic outputs. The smaller the privacy parameters are set, the more noise is injected into the mechanism outputs, entailing a decreasing likelihood of tracing back the instance existence based on outputs.

A commonly used $(\epsilon, 0)$-DP mechanism for real-valued statistical queries is the *Laplace* mechanism. This mechanism injects noise based on a Laplacian distribution with scale $^{\Delta f}/_{\epsilon}$. Δf is called the sensitivity of a statistical query f. Intuitively, Δf indicates the amount of uncertainty we must introduce into the output in order to hide the contribution of single instances at $(\epsilon, 0)$-level. In our context, f is the frequency of a trace variant. Since one individual, i.e., a case, contributes to only one trace, $\Delta f = 1$. In case an individual can appear in more than one trace, the sensitivity needs to be accordingly increased assuming the same value for the privacy parameter ϵ. State-of-the-art event data anonymization frameworks such as our benchmark often use the *Laplace mechanism*.

4 Partition Selection Algorithm

We first highlight the problem of *partition selection* and link it to event data release. Then, the algorithmic details are presented with a brief analysis.

4.1 Partition Selection

Many data analysis tasks can be expressed as per-partition aggregation operations after grouping the data into an unbounded set of partitions. When identifying the variants of a simple log L as categories, the transformation from L to pairs $(\sigma, L(\sigma))$ becomes a specific instance of these aggregation tasks. To render such queries differentially private, two distinct steps need to be executed. First, all aggregation results are perturbed by noise addition of suitable mechanisms. Next, the set of unique partitions must be modified to prevent leakage of information on the true data categories (*differentially private partition selection*) [4,6]. In case of publicly known partitions or bounded partitions from a familiar finite domain, the second step can be reduced to a direct unchanged release or a simple guessing-task, respectively. However, for the most general form of unknown and infinite category domains, guessing is not efficient anymore and an (ϵ, δ)-DP *partition selection* strategy can be used instead.

Recently, in [4], the authors proposed an (ϵ, δ)-DP *partition selection* approach, where they provided a proof of an optimal partition selection rule which maximizes the number of released category-aggregation pairs while preserving (ϵ, δ)-DP. In particular, the authors showed how the aforementioned anonymization steps can be combined into an explicit (ϵ, δ)-DP mechanism based on a k-Truncated Symmetric Geometric Distribution (k-TSGD), see Definition 4. We exploit the analogy between *partition selection* and simple event log publication and transfer this mechanism to the event data context. Definition 5 shows the respective definition based on a k-TSGD.[1]

Definition 4 (k-TSGD). *Given probability* $p \in (0, 1)$, $m = p/(1+(1-p)-2(1-p)^{k+1})$, *and* $k \geq 1$*, the k-TSGD of* (p, k) *over* $\mathbb{Z}$ *formally reads as:*

$$k\text{-}TSGD[X = x \mid p, k] = \begin{cases} m \cdot (1-p)^{|x|} & if x \in [-k, k] \\ 0 & otherwise \end{cases} \tag{1}$$

Definition 5 ((ϵ,δ)-DP for Event Logs Based on k-TSGD). *Let* $\epsilon \in \mathbb{R}_{>0}$ *and* $\delta \in \mathbb{R}_{>0}$ *be the privacy parameters, and* $\mathcal{M}^{k-TSGD}_{\epsilon,\delta} : \mathcal{L} \to \mathcal{L}$ *be a randomized*

[1] A respective proof can be found in Sec. 3 of [4].

mechanism based on a k-TSGD. Given $L \in \mathcal{L}$ *as an input of the randomized mechanism, an event log* $L' = \{(\sigma, L'(\sigma)) \mid \sigma \in L \wedge L'(\sigma) > k\} \in rng(\mathcal{M}^{k-TSGD}_{\epsilon,\delta})$ *is an* (ϵ, δ)*-DP representation of* L *if* $L'(\sigma) = L(\sigma) + x_\sigma$ *is the noisified frequency with* x_σ *being realization of i.i.d random variables drawn from a k-TSGD with parameters* (p, k)*, where* $p = 1 - e^{-\epsilon}$ *and* $k = \lceil 1/\epsilon \times ln((e^\epsilon+2\delta-1)/\delta(e^\epsilon+1)) \rceil$*.*

Definition 5 shows the direct (ϵ, δ)-DP release of trace variants by first perturbing all variant frequencies and then truncating infrequent behavior. Additionally, optimality is guaranteed w.r.t. the number of variants being published due to the *k-TSGD* structure [4]. Note that the underlying *k-TSGD* mechanism assumes each case only contributes to one variant. In case this requirement needs to be violated, sensitivity considerations force a decrease in (ϵ, δ).

The development of differentially private *partition selection* enables significant performance improvements for private trace variant releases. As there are infinite activity sequences defining a variant, former approaches had to either guess or query all of these potentially non-existing sequences in a cumbersome fashion due to the ex-ante category anonymity in (ϵ, δ)-DP. On the contrary, *partition selection* only needs one noisified aggregation operation followed by a specific truncation. Hence, the output contains only existing variants that are independent of external parameters or query patterns.

4.2 Algorithm Design

Algorithm 1 presents the core idea of *TraVaS* which is based on Definition 5. We also propose a utility-aware extension of *TraVaS*, so-called *uTraVaS*, that utilizes the privacy budgets, i.e., ϵ and δ, by several queries w.r.t. data utility. In this paper, we focus on *TraVaS*, the details of *uTraVaS* are provided on GitHub.[2]

Algorithm 1 (TraVaS) allows to anonymize variant-frequency pairs by injecting *k-TSGD* noise within one run over the according simple log. After a simple log L and privacy parameters $(\epsilon > 0, \delta > 0)$ are provided, the *travas* function first transforms (ϵ, δ) into *k-TSGD* parameters (p, k). Then, each variant frequency $L(\sigma)$ becomes noisified using i.i.d *k-TSGD* noise x_σ (see Definition 5). Eventually, the function removes all modified infrequent variants where the perturbed frequencies yield numbers below or equal to k. Due to the partition

Algorithm 1: Differentially Private Trace Variant Selection (TraVaS)

```
Input: Event log $L$, DP-Parameters $(\epsilon, \delta)$
Output: $(\epsilon, \delta)$-DP log $L'$
function travas $(L, \epsilon, \delta)$
    $p = 1 - e^{-\epsilon}$                                              // compute probability
    $k = \lceil 1/\epsilon \times \ln((e^\epsilon + 2\delta - 1)/(\delta(e^\epsilon + 1)))\rceil$   // compute threshold
    forall $(\sigma, L(\sigma)) \in L$ do
        $x_\sigma$ = rTSGD $(p, k)$                                      // generate i.i.d k-TSGD noise
        if $L(\sigma) + x_\sigma > k$ then
            add $(\sigma, L(\sigma) + x_\sigma)$ to $L'$
    return $L'$
```

[2] https://github.com/wangelik/TraVaS/tree/main/supplementary.

selection mechanism, the actual frequencies of all trace variants can directly be queried without guessing trace variants. Thus, *TraVaS* is considerably more efficient and easier to implement than current state-of-the-art prefix-based methods.

5 Experiments

We compare the performance of *TraVaS* against the state-of-the-art benchmark [12] and its extension (*SaCoFa* [9]) on real-life event logs. Due to algorithmic differences between our approach and the prefix-based approaches, it is particularly important to ensure a fair comparison. Hence, we employ divergently structured event logs and study a broad spectrum of privacy budgets (ϵ, δ). Moreover, the sequence cutoff for the benchmark and *SaCoFa* is set to the length that covers 80% of variants in each log, and the remaining pruning parameter is adjusted such that on average anonymized logs contain a comparable number of variants with the original log. Note that *TraVaS* guarantees the optimal number of output variants due to its underlying differentially private partition selection mechanism [4], and it does not need to limit the length of the released variants. Thus, the aforementioned settings consider the limitations of the prefix-based approaches to have a fair comparison.

We select two event logs of varying size and trace uniqueness. As we discussed in Sect. 4, and it is considered in other research such as [9,12], and [14], infrequent variants are challenging to privatize. Thus, trace uniqueness is an important analysis criterion. The Sepsis log describes hospital processes for Sepsis patients and contains many rare traces [11]. In contrast, BPIC13 has significantly more cases at a four times smaller trace uniqueness [5]. The events in BPIC13 belong to an incident and problem management system called VINST. Both logs are realistic examples of confidential human-centered information where the case identifiers refer to individuals. Detailed log statistics are shown in Table 2.

5.1 Evaluation Metrics

To assess the performance of an (ϵ, δ)-DP mechanism, suitable evaluation metrics are needed to determine how valuable the anonymized outputs are w.r.t. the original data. In this respect, we first consider a *data utility* perspective where the similarity between two logs is measured independent of future applications. For our experiments, two respective metrics are considered. From [13], we adopt *relative log similarity* that is based on the *earth mover's distance* between two trace variant distributions, where the normalized *Levenshtein* string edit distance is used as a similarity function between trace variants. The *relative log similarity*

Table 2. General statistics of the event logs used in our experiments.

Event log	#Events	#Cases	#Activities	#Variants	Trace uniqueness
Sepsis	15214	1050	16	846	80%
BPIC13	65533	7554	4	1511	20%

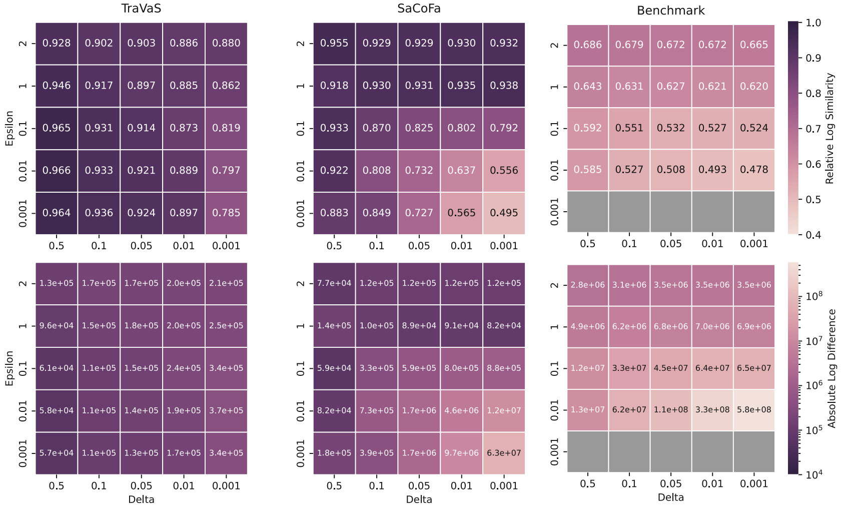

Fig. 1. The *relative log similarity* and *absolute log difference* results of anonymized BPIC13 logs generated by *TraVaS*, the benchmark, and *SaCoFa*. Each value represents the mean of 10 runs.

metric quantifies the degree to which the variant distribution of an anonymized log matches the original variant distribution on a scale from 0 to 1.

In addition, we introduce an *absolute log difference* metric to account for situations where distribution-based metrics provide only different expressiveness. Exemplary cases are event logs possessing similar variant distributions, but significantly different sizes. For such scenarios, the *relative log similarity* yields high similarity scores, whereas *absolute log difference* can detect these size disparities. To derive an absolute log difference value, we first transform both input logs into a *bipartite graph* of variant vertices. Then a *cost network flow* problem [15] is solved by setting demands and supplies to the absolute variant frequencies and utilizing a *Levenshtein* distance between variants as an edge cost. Hence, the resulting optimization value of an (ϵ, δ)-DP log resembles the number of *Levenshtein* operations to transform all respective variants into variants of the original log. In contrast to our *relative log similarity* metric, this approach can also penalize a potential matching impossibility. More information on the exact algorithms is provided on GitHub.[3]

Besides comparing event logs based on *data utility* measures, we additionally quantify the algorithm performance with *process discovery* oriented *result utilities*. We use the *inductive miner infrequent* [10] with default noise threshold of 20% to discover process models from the privatized event logs for all (ϵ, δ) settings under investigation. Then, we compare the models with the original event log to obtain token-based replay *fitness* and *precision* scores [2]. Due to the probabilistic nature of (ϵ, δ)-DP, we average all metrics over 10 anonymized logs for each setting, i.e., 10 separate algorithm runs per setting.

[3] https://github.com/wangelik/TraVaS/tree/main/supplementary.

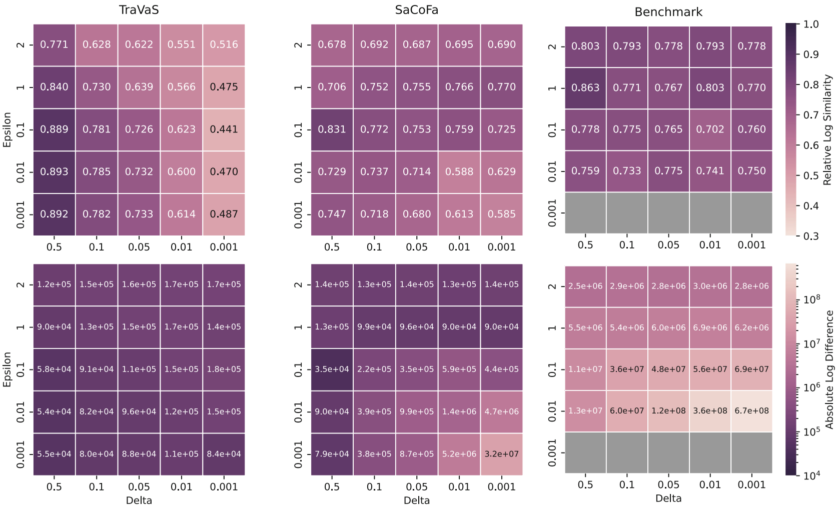

Fig. 2. The *relative log similarity* and *absolute log difference* results of anonymized Sepsis event logs generated by *TraVaS*, the benchmark, and *SaCoFa*. Each value represents the mean of 10 runs.

5.2 Data Utility Analysis

In this subsection, the results of the two aforementioned data utility metrics are presented for both real-life event logs. We compare the performance of *TraVaS* against our benchmark and *SaCoFa* based on the following privacy parameter values: $\epsilon \in \{2, 1, 0.1, 0.01, 0.001\}$ and $\delta \in \{0.5, 0.1, 0.05, 0.01, 0.001\}$.

Figure 1 shows the average results on BPIC13 in a four-fold heatmap. The grey fields represent a general unfeasibility of the strong privacy setting $\epsilon = 0.001$ for our benchmark method. Due to the intense noise perturbation, the corresponding variant generation process ncreased the number of artificial variant fluctuations to an extent that could not be averaged in a reasonable time. Apart from this artifact, both *relative log similarity* and *absolute log difference* show superior performance of *TraVaS* for most investigated (ϵ, δ) combinations. In particular, for stronger privacy settings, *TraVaS* provides a significant advantage over *SaCoFa* and benchmark. Whereas more noise, i.e., lower (ϵ, δ) values, generally decreases the output similarity to the original data, *TraVaS* results seem to particularly depend on δ. According to Definition 5, this observation can be explained by the stronger relation between k and δ compared to k and ϵ.

The evaluation of the Sepsis log is presented in Fig. 2. In contrast to BPIC13, Sepsis contains many variants occurring only once or twice. While our *absolute log difference* shows a similar expected trend with (ϵ, δ) as Fig. 1, the *relative log similarity* metric indicates almost constant values for the prefix-based techniques and a considerable δ-dependency for *TraVaS*. We explain the resulting patterns by examining the underlying data structure in more detail. As mentioned, the frequency threshold k of *TraVaS* strongly correlates with δ. Hence, event logs

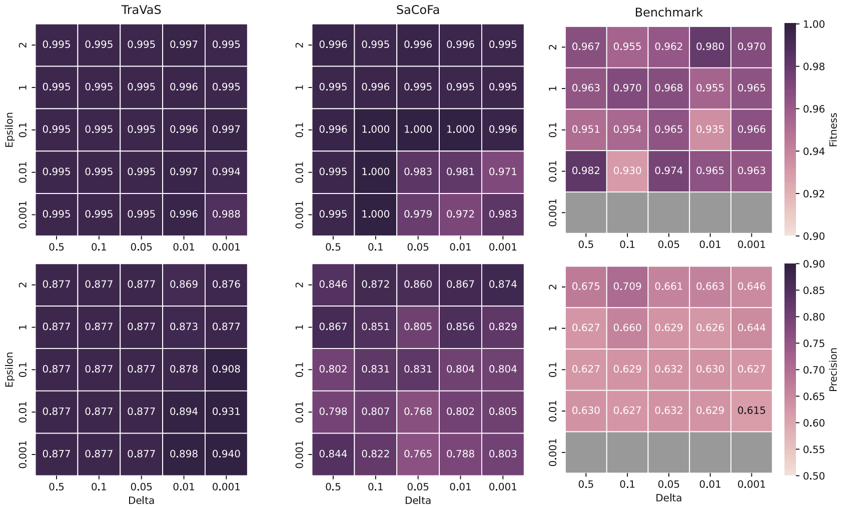

Fig. 3. The *fitness* and *precision* results of anonymized BPIC13 event logs generated by *TraVaS*, the benchmark, and *SaCoFa*. Each value represents the mean of 10 runs.

with prominent infrequent traces, e.g., Sepsis, are significantly truncated for strong (ϵ, δ)-DP. Since this variant removal leads to a distribution mismatch when being compared to the original log, the *relative log similarity* forms a step-wise pattern as in Fig. 2. In contrast, the prefix-based techniques iteratively generate variants that may or may not exist in the original log. In logs with high trace uniqueness, there exist many unique variants that are treated similarly to non-existing variants due to close frequency values, i.e., 0 and 1. Thus, in the anonymized logs, unique variants either appear with larger noisified frequencies or are replaced with fake variants having larger noisified frequencies. This process remains the same for different privacy settings but with larger frequencies for stronger privacy guarantees. Hence, the *relative log similarity* metric stays almost constant although the noise increases with stronger privacy settings. However, the *absolute log difference* metric can show differences. *uTraVaS* shows even better performance w.r.t. the data utility metrics.[4]

5.3 Process Discovery Analysis

We conduct a *process discovery* investigation based on *fitness* and *precision* scores. For the sake of comparability, the experimental setup remains unchanged. Figure 3 shows the results for BPIC13, where the original fitness and precision values are 0.995 and 0.877, respectively. *TraVaS* provides almost perfect replay behavior w.r.t. *fitness* while the prefix-based alternatives show lower values. This observation can be explained by the different algorithmic approach of *TraVaS* and some characteristics of BPIC13. *TraVaS* only adopts true behavior that

[4] https://github.com/wangelik/TraVaS/tree/main/experiments.

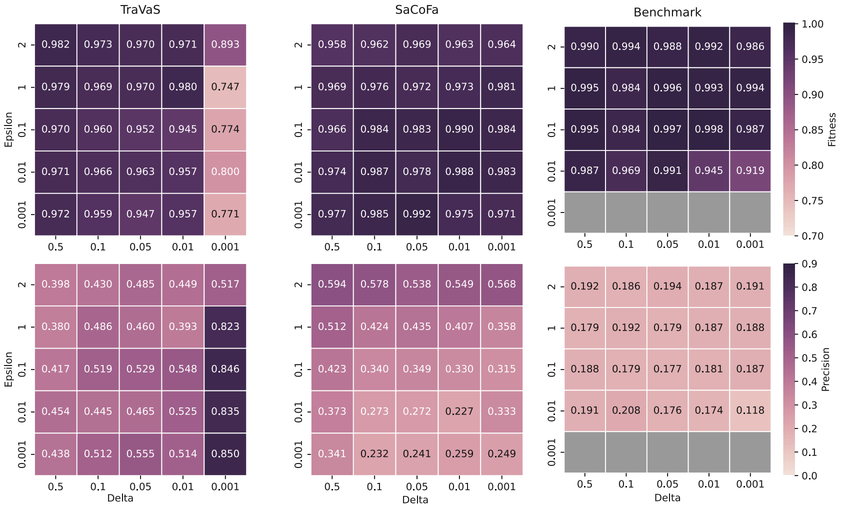

Fig. 4. The *fitness* and *precision* results of anonymized Sepsis event logs generated by *TraVaS*, the benchmark, and *SaCoFa*. Each value represents the mean of 10 algorithm runs.

results in a simplified representation of the original process model. Due to the rather low trace uniqueness and comparably large log-size of BPIC13, this simplification is minor enough to allow an almost perfect fitness. In contrast, the fake variants generated by prefix-based approaches negatively affect their fitness scores. The precision metric evaluates the fraction of behavior in a model discovered from an anonymized log that is not included in the original log. Due to the direct release mechanism of *TraVaS* that only removes infrequent variants, we achieve more precise process models than the alternatives. Furthermore, the correlation between threshold k and noise intensity enables *TraVaS* to even rise precision for stronger privacy guarantees. Conversely, the fake variants generated by prefix-based approaches can lead to inverse behavior.

Figure 4 shows the *fitness* and *precision* results for Sepsis, where the original fitness and precision values are 0.952 and 0.489, respectively. Whereas *TraVaS* dominates the prefix-based approaches w.r.t. *precision* as in Fig. 3, our fitness score shows a slight under-performance. Unlike BPIC13, the high trace uniqueness and smaller log-size prohibit the underlying *partition selection* mechanism to achieve negligible threshold for infrequent variant removal. Thus, the discovered process models from anonymized logs miss parts of the original behavior. This shows that carefully tuned prefix-based mechanisms might have an advantage in terms of fitness for small logs with many unique traces. We particularly note that this limitation of *TraVaS* vanishes as soon as the overall log-size grows. The reason lies in the size-independent threshold k while the pruning parameter of prefix-based approaches intensifies with the data size. The process discovery analyses for *uTraVaS*, available on GitHub, show even better performance.

6 Discussion and Conclusion

In this paper, we demonstrated a novel approach to release anonymized distributions of trace variants based on (ϵ, δ)-DP mechanisms. The corresponding algorithm (*TraVaS*) overcomes the variant generation problems of prefix-based mechanisms (see Sect. 1) and directly queries all true variants. Our experiments with two differently structured event logs showed that *TraVaS* outperforms the state-of-the-art approaches in terms of *data utility* metrics and process-discovery-based *result utility* for most of the privacy settings. In particular, for large event logs containing many long trace variants, our implementation has no efficient alternative. Regarding limitations and future improvements, we generally note that the differentially private partition selection mechanism only works for $\delta{>}0$, whereby limits of small values can be problematic on large collections of infrequent variants. Thus, all use cases that require strict ϵ-DP still need to apply prefix-based mechanisms. Finding a more efficient solution for $\delta = 0$ seems to be a valuable and interesting future research topic.

References

1. GDPR. https://data.europa.eu/eli/reg/2016/679/oj. Accessed 15 May 2021
2. van der Aalst, W.M.P.: Process Mining - Data Science in Action, 2nd edn. Springer, Cham (2016)
3. Cohen, A., Nissim, K.: Towards formalizing the GDPR's notion of singling out. Proc. Natl. Acad. Sci. U.S.A. **117**(15), 8344–8352 (2020)
4. Desfontaines, D., Voss, J., Gipson, B., Mandayam, C.: Differentially private partition selection. Proc. Priv. Enhancing Technol. **2022**(1), 339–352 (2022)
5. van Dongen, B.F., Weber, B., Ferreira, D.R., Weerdt, J.D.: BPI challenge 2013. In: Proceedings of the 3rd Business Process Intelligence Challenge (2013)
6. Dwork, C.: Differential privacy: a survey of results. In: Agrawal, M., Du, D., Duan, Z., Li, A. (eds.) TAMC 2008. LNCS, vol. 4978, pp. 1–19. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-79228-4_1
7. Elkoumy, G., Pankova, A., Dumas, M.: Privacy-preserving directly-follows graphs: balancing risk and utility in process mining. CoRR abs/2012.01119 (2020)
8. Fahrenkrog-Petersen, S.A., van der Aa, H., Weidlich, M.: PRIPEL: privacy-preserving event log publishing including contextual information. In: Fahland, D., Ghidini, C., Becker, J., Dumas, M. (eds.) BPM 2020. LNCS, vol. 12168, pp. 111–128. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58666-9_7
9. Fahrenkrog-Petersen, S.A., Kabierski, M., Rösel, F., van der Aa, H., Weidlich, M.: Sacofa: semantics-aware control-flow anonymization for process mining. In: 3rd International Conference on Process Mining, ICPM. IEEE (2021)
10. Leemans, S.J.J., Fahland, D., van der Aalst, W.M.P.: Discovering block-structured process models from event logs containing infrequent behaviour. In: Lohmann, N., Song, M., Wohed, P. (eds.) BPM 2013. LNBIP, vol. 171, pp. 66–78. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-06257-0_6
11. Mannhardt, F.: Sepsis Cases (2016). https://doi.org/10.4121/uuid:915d2bfb-7e84-49ad-a286-dc35f063a460
12. Mannhardt, F., Koschmider, A., Baracaldo, N., Weidlich, M., Michael, J.: Privacy-preserving process mining - differential privacy for event logs. Bus. Inf. Syst. Eng. **61**(5), 595–614 (2019)

13. Rafiei, M., van der Aalst, W.M.P.: Towards quantifying privacy in process mining. In: Leemans, S., Leopold, H. (eds.) ICPM 2020. LNBIP, vol. 406, pp. 385–397. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-72693-5_29
14. Rafiei, M., van der Aalst, W.M.P.: Group-based privacy preservation techniques for process mining. Data Knowl. Eng. **134**, 101908 (2021)
15. Tomlin, J.A.: Minimum-cost multicommodity network flows. Oper. Res. **14**, 45–51 (1966)