A Generalizable Approach for Determining The Sensitivity of A Trace within An Event Log

Ryan Hildebrant, Zhenyu Zhang, and Shangping Ren
Department of Computer Science, San Diego State University
{rhildebrant, zzhang, sren}@sdsu.edu

Abstract. In this Research-In-Progress work, we present a potentially generalizable approach that can determine the sensitivity of an attribute or a set of attributes associated with an event in a given event log. This approach is based on the concept of an equivalence class that a given event or trace may form and associate its sensitivity with the size of its equivalence class. For a given event, different equivalent classes can be formed based on different attributes, the proposed approach provides researchers with a more granular tool to apply group based privacy to event logs.

Keywords: K-anonymity Privacy · Process Mining

1 Introduction

Privacy in process mining has been cited as a critical component for process mining applications [1]. Various researchers have presented work that focuses on group-based anonymization techniques and recognize the importance of attributes linked to individual events [2] [3]. However, all of their work focuses on either events or traces and define unique notations to support their approaches, respectively. Because of this, it is difficult to determine the appropriated approach for a given event log. It also makes it more challenging to integrate multiple approaches that apply privacy at different levels. In contrast, we present a generalizable approach that is based on the equivalence classes that a given event may form and associate sensitivities with the size of the equivalent classes.

2 Determining Equivalent Classes and Sensitivity

We use an example to explain for a given event log, how different equivalent classes may be formed and how the event sensitives are related to the cardinalities of the equivalent classes. Assume we have the following event log, where each event entry is represented by `<eventId, traceId, actName, [attr1, ···, attrn]>`. We use * to denote the wildcard which can be of any value.

eventID	traceID	actName	Attributes
e1	1	print	[John, 2:00pm, scripps, *, *]
e2	1	deposit	[Brad, 2:01pm, scripps, $100, Teller]
e3	1	print	[Brad, 2:04pm, scripps, *, *]
e4	2	withdraw	[Alice, 8:00am, SDSU, $200, Teller]
e5	3	withdraw	[Alice, 8:00am, SDSU, $200, Teller]
e6	4	print	[Bob, 2:00pm, scripps, *, *]
e7	4	deposit	[Alice, 2:01pm, scripps, $100, ATM]
e8	4	print	[John, 2:04pm, scripps, *, *]
e9	5	print	[John, 2:00pm, scripps, *, *]
e10	5	deposit	[John, 2:01pm, scripps, $100, ATM]
e11	5	print	[John, 2:04pm, scripps, *, *]

For the given event log, we can form different equivalence classes based on event attribute relations and traces. For instance, consider the event $< print >$. We can form an equivalent class that has the same attribute value of `John` for the `customer` attribute and `scripps` for the `location`. In this case, the equivalent class contains four elements, i.e., $e1, e8, e9$, and $e11$. We can also form an equivalent class that only considers the `location` attribute `scripps`, which has six elements, i.e., $e1, e3, e6, e8, e9$, and $e11$. Therefore, the sensitivity of `John` in terms of his activity location of "scripts" is 4/6. Additionally, we can compare the other attribute values with respect to the event performed by `Bob` and `Brad`, respectively, which is $\frac{1}{6}$. By comparing all of these values, we can conclude that `Bob` and `Brad` are more sensitive than `John` in terms of their activity locations.

This notion of equivalent class and sensitivity can also be applied to traces. Consider traces σ_1, σ_4, and σ_5. If we define our equivalent class on traces that have the same attributes values as `timestamp`, `location`, and `amount`, the three traces then belong to the same equivalent class. In fact, for the given event log, the cardinality of the equivalent class that has the same values of `timestamp`, the `location` and the `amount` is three. We can form an even tighter relation on a subset of these traces, i.e., requiring *timestamp, location, amount, provider* values to be the same, which result in an equivalent class with only two elements, i.e., $\{\sigma_4, \sigma_5\}$. Hence, the sensitivity of `provider` information in a trace with respect to *timestamp, location, amount* is $\frac{2}{3}$. On the other hand, we can change the `provider` value to $[Teller]$ and get a more sensitive result, $\frac{1}{3}$. This result is so sensitive that we can single out a trace within this log. Therefore, an attacker who knows these specific attribute values combinations of `timestamp`, `location`, `amount`, and `provider` could directly identify a set of individuals associated to σ_1.

3 Conclusion

We have briefly shown how we can use equivalent classes to define groups and provided a uniform representation that supports techniques to improve group-based privacy. Our future work is to formalize the notation and apply it to existing work, such as [2] and [3], and compare the results in terms of sensitivity (which quantitatively measures privacy). In addition, we will study how to modify raw event logs to maximize the sensitivity of all information in the log and at the same time minimize the impact on the structural change of event model resulted from the modified log.

References

1. Wil Van Der Aalst. Data science in action. In *Process mining*, pages 3–23. Springer, 2016.
2. Stephan A Fahrenkrog-Petersen, Han van der Aa, and Matthias Weidlich. Pretsa: event log sanitization for privacy-aware process discovery. In *2019 International Conference on Process Mining (ICPM)*, pages 1–8. IEEE, 2019.
3. Majid Rafiei, Miriam Wagner, and Wil MP van der Aalst. Tlkc-privacy model for process mining. In *International Conference on Research Challenges in Information Science*, pages 398–416. Springer, 2020.