Traceability in Process Analysis

Maike Basmer[1]

[1]*Humboldt-Universität zu Berlin, Unter den Linden 6, 10099 Berlin, Germany*

Abstract

The exploratory nature of process analysis requires the analysts to make decisions not only during the analysis but also during data preparation, which affects the outcome of the analysis. This PhD project aims to support traceability in process analysis, i.e., reconstructing the effect of the input data and the decisions made throughout the process analysis pipeline on the outcome. To accomplish this, we plan to leverage established data management capacities to integrate the models used for pre-processing and analysis.

Keywords

process analysis, traceability, database systems

1. Motivation

Process analysis often follows an exploratory approach: while scrutinizing the event data captured from a process, analysts continuously build hypotheses and subsequently seek to falsify or validate them based on the data. This potentially involves comparing different process mining algorithms or testing different parameters. However, the exploration of the event data does not just start with the analysis, but rather when the data is prepared, as the data extraction, transformation, and loading (ETL) may also be subject to frequent change. Thus, decoupling the analysis from the data preparation possibly hides the effect of the choices made during the ETL steps on the analysis outcome. That does not only make it difficult to relate the results of the analysis to the original data, but also to judge the reliability of the results at large. Database technology appears to have the means in store to address that challenge, as they allow to integrate the ETL process and analysis using unified data models and query languages. Instead of extracting the event data to a log, one may keep the data close to the source, thus allowing to trace the analysis results to the source data. Ultimately, this affords the opportunity to reason on the propagation or interplay of changes in the pre-processing phase with respect to the analysis. That way, process analysts are supported in tracking and understanding the impact of decisions made during data preparation and analysis, which enables them to justify those decisions. Furthermore, adopting standard data models and query languages as the basis for this integration enables us to leverage the capacity of database systems and the research on them spanning decades to support the process analysis. Accordingly, the complex of problems that is going to be addressed in the PhD thesis can be summarized as follows:

Objective of PhD Project

We aim at achieving traceability in process analysis, i.e., the ability to trace analysis results to the input data. To accomplish this, we will integrate the models used for data pre-processing and data analysis, and operationalize them using existing data management capacities.

ICPM 2024 Doctoral Consortium, October 14–18, 2024, Kongens Lyngby, Denmark

maike.basmer@hu-berlin.de (M. Basmer)
0000-0003-2124-0276 (M. Basmer)

2. Related Work

2.1. Database Technology in Process Mining

Within the relational realm, intermediate in-database representations and a native in-database operator have been developed to accelerate process mining tasks [1, 2]. Furthermore, concepts from data warehousing were adopted to facilitate multidimensional analysis [3, 4]. Schönig et al. [5, 6] implemented declarative process discovery on relational databases. Riva et al. [7] considered different schemata that have been proposed to represent event logs in the past and examined the effect of the schema choice on the performance of declarative process mining. Besides that, modelling event data as labeled property graphs [8] was proposed to enable graph-based understanding of multi-dimensional data and to accommodate different analyses [9].

2.2. Supporting the Process Analysis Pipeline

A process analysis pipeline may encompass different pre-processing steps like integrating, transforming, reducing, abstracting, filtering, or enriching the event log before the analysis [10], with abstraction currently being the focus for our setting. Different approaches to event abstraction exist, mainly lifting low-level events to activities according to the domain [11, 12]. Other types of high-level events may also be discovered to enhance the analysis of processes [9, 13].

Regarding traceability, there have been several proposals in the past. Probabilistic event abstraction allows to keep track of alternative abstractions by capturing uncertainty when producing high-level events [14]. For process mining on IoT data, Bertrand et al. [15] propose a schema for an event log that caters to traceability concerns as well as different needs in granularity. Klinkmüller et al. [16] examine the sensitivity of discovery results w.r.t. pipeline operations and parameters to debug process discovery pipelines, encompassing the discovery procedure itself along with pre-processing steps like abstraction or filtering. Data and provenance views were proposed to support explorative process mining by tracking steps, goals, and intermediate results throughout the analysis process [17]. Beyond process mining, further inspiration may be drawn from research on provenance [18], explanations [19], debugging of pipelines [20], or probabilistic databases [21].

3. Overview of Research Project

In the course of the PhD project, several facets may be investigated, for example:

- Which data schema or data model should be used depending on the use case or the characteristics of the data?
- Can we exploit properties of the data to support the process analysis?

We will focus on two specific use cases described below to grasp and better understand these questions and the arising challenges.

3.1. Realization

3.1.1. Tracing the Effect of Abstractions

To target the traceability of abstractions and their effect on a given analysis, the concept of Event Knowledge Graphs (EKGs) [8] implemented in graph databases [22] may come in handy, as they integrate low-level events with high-level abstractions and enable graph-based querying. This capacity may be extended to record event abstractions, such that the effect of abstractions during exploratory process analysis can be tracked. To that end, we conceive the following framework: In a forward-manner, the abstractions represented as queries in a given data preparation pipeline are treated as first-class citizens of an EKG by recording them along with their relations to lower-level events. Considering an alternative abstraction in the pipeline, the intermediary results of that alternative pipeline are computed and recorded correspondingly. Differences in the analysis may be explained by the difference set of nodes or edges between both possible “worlds” - either by their mere (non-)existence in one set or the other or by the context they define (i.e., the features that distinguish those nodes or edges). We plan to apply this idea to a pipeline for task analysis [23], as it involves several steps of abstraction. Interaction mining [24] may also lend itself to evaluating this idea.

3.1.2. Multi-Dimensional Declarative Process Mining in Relational Databases

Similarly, the rich feature set of relational database systems may be employed to host process mining tasks. We plan to focus on declarative process mining [25], especially in view of multiple dimensions [26, 27], as data-aware conditions relate to selection and navigating relations correspond to joins in the relational model. Implementing conformance checking or process discovery for multi-dimensional declarative process specifications encompasses finding an adequate representation of the event data, encoding the task as a set of queries, and ideally leveraging database technology like materialized views [28] to track and reuse intermediary results. Another aspect that could be exploited in case of declarative process specifications is their apparent similarity to data dependencies in relational databases. In that case, techniques from the domain of data profiling may be used as a basis for, e.g., the discovery of declarative constraints [29]. Beyond that, it might be interesting to investigate which intermediary data representations like indices [30] or materialized views [28] or other developments from database systems research like row pattern recognition [31] may be useful to realize process mining tasks in-database.

3.2. Evaluation

Developments aiming at enhancing the efficiency of process analysis tasks may be evaluated empirically in a set of experiments on data sets that are established within the process mining community. In addition to that, synthetic data may serve to investigate the influence of specific data properties on the interventions that are going to be devised during the PhD project. When it comes to evaluating the traceability, one can either head into the direction of showing that the developed approach fulfills certain properties or measure the capacity of the proposed approach to trace deviations in the analysis due to abstractions. For example, it might be sensible to measure how compact these insights can be represented if we assume a correlation between the compactness of the representation and understandability.

4. Conclusion

The proposed thesis sets out to integrate pre-processing of the event data with process analysis by means of database technologies to achieve traceability. We outlined ideas how to approach this problem set, e.g., through the lens of event knowledge graphs (in terms of database technology used) or declarative process mining (in terms of process analysis).

References

[1] A. Syamsiyah, S. J. J. Leemans, Process discovery using in-database minimum self distance abstractions, in: Proceedings of the 35th Annual ACM Symposium on Applied Computing, ACM, Brno Czech Republic, 2020, pp. 26–35. doi:10.1145/3341105.3373846.

[2] R. Dijkman, J. Gao, A. Syamsiyah, B. van Dongen, P. Grefen, A. ter Hofstede, Enabling efficient process mining on large data sets: Realizing an in-database process mining operator, Distributed and Parallel Databases 38 (2020) 227–253. doi:10.1007/s10619-019-07270-1.

[3] W. M. P. Van Der Aalst, Process Cubes: Slicing, Dicing, Rolling Up and Drilling Down Event Data for Process Mining, in: Asia Pacific Business Process Management, volume 159, Springer, 2013, pp. 1–22. doi:10.1007/978-3-319-02922-1_1.

[4] T. Vogelgesang, H.-J. Appelrath, PMCube: A Data-Warehouse-Based Approach for Multidimensional Process Mining, in: Business Process Management Workshops, LNBIP, Springer, 2016, pp. 167–178. doi:10.1007/978-3-319-42887-1_14.

[5] S. Schönig, C. Di Ciccio, F. M. Maggi, J. Mendling, Discovery of Multi-perspective Declarative Process Models, in: Service-Oriented Computing, volume 9936, Springer, 2016, pp. 87–103. doi:10.1007/978-3-319-46295-0_6.

[6] S. Schönig, A. Rogge-Solti, C. Cabanillas, S. Jablonski, J. Mendling, Efficient and Customisable Declarative Process Mining with SQL, in: Advanced Information Systems Engineering, volume 9694, Springer, 2016, pp. 290–305. doi:10.1007/978-3-319-39696-5_18.

[7] F. Riva, D. Benvenuti, F. M. Maggi, A. Marrella, M. Montali, An SQL-Based Declarative Process Mining Framework for Analyzing Process Data Stored in Relational Databases, in: Business Process Management Forum, volume 490, Springer Nature Switzerland, 2023, pp. 214–231. doi:10.1007/978-3-031-41623-1_13.

[8] S. Esser, D. Fahland, Multi-Dimensional Event Data in Graph Databases, Journal on Data Semantics 10 (2021) 109–141. doi:10.1007/s13740-021-00122-1.

[9] E. L. Klijn, F. Mannhardt, D. Fahland, Aggregating Event Knowledge Graphs for Task Analysis, in: Process Mining Workshops, LNBIP, Springer Nature Switzerland, 2023, pp. 493–505. doi:10.1007/978-3-031-27815-0_36.

[10] Y. Liu, V. S. Dani, I. Beerepoot, X. Lu, Turning logs into lumber: Preprocessing tasks in process mining, in: J. D. Smedt, P. Soffer (Eds.), Process Mining Workshops - ICPM 2023 International Workshops, Rome, Italy, October 23-27, 2023, Revised Selected Papers, volume 503 of *LNBIP*, Springer, 2023, pp. 98–109. doi:10.1007/978-3-031-56107-8_8.

[11] S. J. Van Zelst, F. Mannhardt, M. De Leoni, A. Koschmider, Event abstraction in process mining: Literature review and taxonomy, Granular Computing 6 (2021) 719–736. doi:10.1007/s41066-020-00226-2.

[12] K. Diba, K. Batoulis, M. Weidlich, M. Weske, Extraction, correlation, and abstraction of event data for process mining, WIREs Data Mining and Knowledge Discovery 10 (2020) e1346. doi:10.1002/widm.1346.

[13] B. Bakullari, J. van Thoor, D. Fahland, W. M. P. van der Aalst, The Interplay Between High-Level Problems and The Process Instances That Give Rise To Them, 2023. arXiv:2309.01571.

[14] B. Fazzinga, S. Flesca, F. Furfaro, E. Masciari, L. Pontieri, Efficiently interpreting traces of low level events in business process logs, Information Systems 73 (2018) 1–24. doi:10.1016/j.is.2017.11.001.

[15] Y. Bertrand, S. Veneruso, F. Leotta, M. Mecella, E. Serral, NICE: The Native IoT-Centric Event Log Model for Process Mining, in: LNBIP, Springer Verlag (Germany), Rome, 2023.

[16] C. Klinkmüller, A. Seeliger, R. Müller, L. Pufahl, I. Weber, A Method for Debugging Process Discovery Pipelines to Analyze the Consistency of Model Properties, in: Business Process Management, volume 12875, Springer, 2021, pp. 65–84. doi:10.1007/978-3-030-85469-0_7.

[17] F. Zerbato, A. Burattin, H. Völzer, P. N. Becker, E. Boscaini, B. Weber, Supporting Provenance and Data Awareness in Exploratory Process Mining, in: Advanced Information Systems Engineering, volume 13901, Springer Nature Switzerland, 2023, pp. 454–470. doi:10.1007/978-3-031-34560-9_27.

[18] B. Glavic, Data Provenance, Foundations and Trends® in Databases 9 (2021) 209–441. doi:10.1561/1900000068.

[19] B. Glavic, A. Meliou, S. Roy, Trends in Explanations: Understanding and Debugging Data-driven Systems, Foundations and Trends® in Databases 11 (2021) 226–318. doi:10.1561/1900000074.

[20] R. Lourenço, J. Freire, E. Simon, G. Weber, D. Shasha, BugDoc, The VLDB Journal 32 (2023) 75–101. doi:10.1007/s00778-022-00733-5.

[21] D. Suciu, Probabilistic databases, in: Encyclopedia of Database Systems, Second Edition, Springer, 2018. doi:10.1007/978-1-4614-8265-9_275.

[22] Graph Data Management: Fundamental Issues and Recent Developments, Data-Centric Systems and Applications, Springer, 2018. doi:10.1007/978-3-319-96193-4.

[23] E. L. Klijn, F. Mannhardt, D. Fahland, Multi-perspective concept drift detection: Including the actor perspective, in: Advanced Information Systems Engineering - 36th International Conference, CAiSE 2024, Limassol, Cyprus, June 3-7, 2024, Proceedings, volume 14663 of *Lecture Notes in Computer Science*, Springer, 2024, pp. 141–157. doi:10.1007/978-3-031-61057-8_9.

[24] A. Senderovich, A. Rogge-Solti, A. Gal, J. Mendling, A. Mandelbaum, The ROAD from Sensor Data to Process Instances via Interaction Mining, in: S. Nurcan, P. Soffer, M. Bajec, J. Eder (Eds.), Advanced Information Systems Engineering, Lecture Notes in Computer Science, Springer International Publishing, Cham, 2016, pp. 257–273. doi:10.1007/978-3-319-39696-5_16.

[25] C. Di Ciccio, M. Montali, Declarative Process Specifications: Reasoning, Discovery, Monitoring, in: Process Mining Handbook, Springer, 2022, pp. 108–152. doi:10.1007/978-3-031-08848-3_4.

[26] A. Burattin, F. M. Maggi, A. Sperduti, Conformance checking based on multi-perspective declarative process models, Expert Systems with Applications 65 (2016) 194–211. doi:10.1016/j.eswa.2016.08.040.

[27] W. M. P. van der Aalst, G. Li, M. Montali, Object-Centric Behavioral Constraints, 2017. arXiv:1703.05740.

[28] R. Shirkova, J. Yang, Materialized Views, Foundations and Trends® in Databases 4 (2011) 295–405. doi:10.1561/1900000020.

[29] X. Chu, I. F. Ilyas, P. Papotti, Discovering denial constraints, Proceedings of the VLDB Endowment 6 (2013) 1498–1509. doi:10.14778/2536258.2536262.

[30] T. Kraska, A. Beutel, E. H. Chi, J. Dean, N. Polyzotis, The Case for Learned Index Structures, in: Proceedings of the 2018 International Conference on Management of Data, ACM, Houston TX USA, 2018, pp. 489–504. doi:10.1145/3183713.3196909.

[31] D. Petković, Specification of Row Pattern Recognition in the SQL Standard and its Implementations, Datenbank-Spektrum 22 (2022) 163–174. doi:10.1007/s13222-022-00404-3.