EMISA 1 Wednesday, 11:00 - 12:30
Room: Potsdam | Session Chair: Ralf Laue

A survey of approaches for event sequence analysis and visualization (Extended Abstract)
Anton Yeshchenko and Jan Mendling

Integrating Fairness into Process Mining Algorithms (PhD Proposal)
Rachmadita Andreswari

EMISA 2 Wednesday, 14:00 - 15:30
Room: Potsdam | Session Chair: Stephan Fahrenkrog-Petersen

Towards a Threat Modeling Language for Vessel Navigation and Port Call Optimization: harborLang (Full Paper)
Simon Hacks

Domain-Specific Conceptual Modeling: An instrument for increasing productivity in system development (Extended Abstract)
Martin Paczona, Heinrich C. Mayr and Guenter Prochart

Investigating the impact of representation features on decision model comprehension (Extended Abstract)
Djordje Djurica, Tyge F. Kummer, Jan Mendling and Kathrin Figl

EMISA 3 Thursday, 11:00 - 12:30
Room: Potsdam | Session Chair: Stephan Fahrenkrog-Petersen

Process Mining Meets Visual Analytics: The Case of Conformance Checking (Extended Abstract)
Luise Pufahl, Michael Grohs, Lisa-Marie Klein and Jana-Rebecca Rehse

Describing Behavior Sequences of Fattening Pigs Using Process Mining on Video Data and Automated Pig Behavior Recognition (Extended Abstract)

Arvid Lepsien, Andreas Melfsen, Jan Bosselmann, Agnes Koschmider and Eberhard Hartung

Addressing the Log Representativeness Problem using Species Discovery (Extended Abstract)
Martin Kabierski, Markus Richter and Matthias Weidlich

A survey of approaches for event sequence analysis and visualization (Extended Abstract)

Anton Yeshchenko [1] and Jan Mendling [1,2,3]

Abstract: Event sequence analysis is an important field of computer science due to its relevance to a diverse spectrum of application domains such as manufacturing, logistics, healthcare, financial services, education [1], to name but a few. Despite this broad relevance across these domains, it is striking to observe that techniques for event sequence data analysis have been developed rather independently in different fields of computer science.

The most prominent research fields investigating the analysis of event sequence data are process mining and information visualization. Process mining has emerged as a subfield of research on workflow management systems [2]. Its focus is the development of new techniques for automatic process discovery from event sequence data with the ambition to provide a meaningful and understandable summary of the behavior to the business process analyst. Information visualization is a field of computer graphics, which originated as a subfield of human–computer interaction [3]. Its focus is on devising new techniques for visualizing event sequence data in a meaningful way such that analysts can effectively explore them. Typical representations frequently used in this field are timelines that plot conceptually related sequences of events over a time axis. As similar as the ambitions of these research areas may sound, it is surprising that there is hardly any exchange of ideas. Cross-references are scarce and mutual awareness and understanding is limited.1 All this makes research on event sequence analysis a fragmented field with scattered contributions.

So far, the contributions from these two fields have neither been compared nor have they been mapped to an integrated framework. At this stage, it is not clear to which extent both fields have developed complementary concepts and insights. Such intransparency is problematic since it bears the risk of opportunities of integration are missed and concepts established in one field are independently reinvented in the other one. In this current research talk at EMISA 2024 based on a recent article published in [YM24], we develop such a framework that we call Event Sequence Visualization framework (ESeVis) and that gives due credit to the traditions of both fields. Our mapping study provides an integrated perspective on both fields and potential synergies for future research. In this way, our work contributes towards overcoming the fragmentation of research on event sequence data analysis.

Keywords: Event sequence data, Information visualization, Process mining

Bibliography

[YM24] Yeshchenko, Anton; Mendling, Jan: A survey of approaches for event sequence analysis and visualization. Inf. Syst., 120:102283, 2024.

1 Vienna University of Economics and Business, 1020 Wien, Österreich, anton.yeshchenko@wu.ac.at, https://orcid.org/0000-0000-0000-0000

2 Humboldt-Universität zu Berlin, 10099 Berlin, Deutschland,

3 Weizenbaum-Institut, 10623 Berlin, Deutschland,

PhD Proposal: Integrating Fairness into Process Mining Algorithms*

Rachmadita Andre Swari[1,2,*,†]

[1]Humboldt Universitat zu Berlin, Unter den Linden 6, Berlin, 10117, Germany
[2]Telkom University, Jl. Telekomunikasi no.1, Bandung, 40257, Indonesia

Abstract

Process mining techniques provide operational insights into work processes in various types of organizations. These processes handle sensitive data of customers, patients, students, or citizens and their results impact the lives and careers of these affected persons. So far, much of process mining research has focused on classical dimensions of performance such a cycle time or operational cost. What is missing is a primal consideration of ethical concerns such as fairness. Fairness is a recently researched concept in machine learning, which requires a deeper integration into process mining algorithms. This research addresses this requirement. To this end, it aims to analyze fairness concerns in process mining, to develop new process mining algorithms that integrate fairness concerns, and to evaluate them for their effectiveness. Methodologically, our research will build on guidelines for design science and algorithm engineering research. In this way, we will combine engineering research with empirical evaluations.

Keywords

process mining, fairness, algorithm, work process

1. Introduction

Process mining is a research field that develops algorithms and analysis techniques for generating insights into work processes based on event log data from information systems that support individual process executions. According to van der Aalst [1], there are three categories of process mining techniques, namely discovery, conformance checking, and enhancement. In industrial projects involving process mining, the starting point is often the *discovery* of a process model. Conformance checking can be used to check if reality, as recorded in the log, conforms to the model and vice versa. Enhancement improves an existing process model using information about the actual process recorded in some event log. Applications of process mining have been reported in various industries, such as healthcare [2], audit [3], sales [4], education [5], software development [6], bank [7], etc. Process mining is appreciated by process analysts for its capability to provide insights into what really happens based on arguably objective event

EMISA'24: 14th International Workshop on Enterprise Modeling and Information Systems Architecture, March 13–14, 2024, Potsdam, Germany

*You can use this document as the template for preparing your publication. We recommend using the latest version of the ceurart style.

*Corresponding author.
†These authors contributed equally.
rachmadita.andre.swari@hu-berlin.de (R. A. Swari)
0000-0001-7104-4746 (R. A. Swari)

logs that are generated through the information system. By this approach, stakeholders received useful information to improve their business process [8] and outcomes [9].

So far, much of process mining research has focused on classical dimensions of performance such a cycle time or operational cost. What is missing is a primal consideration of ethical concerns such as fairness. Analysis of fairness, however, requires that sensitive attributes are preserved. These attributes can then be taking into account for assessing fairness such as race, color, religion, national origin, gender identity, marital status, and age. Human decisions are subjective, and they often include biases. Therefore, there is a widespread belief that employing an automated algorithm enhances the objectivity or fairness of decisions [10]. Some examples of fairness can be examined in the criminal justice system, recruitment system, lending system, education, healthcare system, and also bureaucracy. Fairness, a concept that has recently integrated attention in machine learning [11], needs to be more deeply incorporated into process mining algorithms. This study aims to meet this need by identifying unfairness throughout business processes. The proposal is organized as follows: in Section 2, we discuss existing research; in Section 3, we present the challenges to be addressed in this research; in Section 4, we provide the research design; and finally, in the last section, we conclude the overall purpose of this research.

2. Existing Research

Development of fairness-aware algorithm in machine learning. Fairness-aware algorithms have been widely discussed in the context of machine learning research [12], [13], and [14]. These works highlight four main categories of causes to algorithmic unfairness: biases in the data set, biases caused by missing data, biases that stem from algorithmic objectives, and biases resulting from the use of proxy variables [10]. Biases already included in the data sets used for learning, biases caused by missing data encompassing instances of missing values or sample/selection biases, and algorithmic biases emerge from objectives aimed at minimizing overall aggregated prediction errors, resulting in preferential treatment of majority groups over minorities. To ensure clear fairness metrics, it involves pre-processing techniques and rule-based approaches. However, this approach has limitations. Pre-processing techniques may depend on general fairness measurements, potentially overlooking the unique complexities of the processes. Similarly, rule-based approaches might oversimplify nuanced fairness concerns and struggle to adapt to the dynamic nature of the contexts.

Fairness-awareness in the context of process mining. Process mining research so far only partially reflects these concepts. Some first papers on fairness-aware process mining are [15], [16], [17]. In the study [15], demographic parity was used as the main concept to measure discrimination in the data. The analysis is conceptualized using decision trees as classifiers. In process mining, study related to fairness defined two entities, resource and case [16]. Moreover, [17] identified eight outcomes that can be considered for analyzing business process fairness: time, re-do's, success, deviation, decisions, resource allocation, workload of resource and representation. The measurement results are then applied at the level of individual fairness, group fairness and resource fairness.

3. Challenges to be Addressed

Several challenges in this research should be considered. Firstly, the initial case study for this research will analyze loan applications. Some fairness research has been conducted since discrimination in loan applications has been identified [18] [19] and [20]. Subsequently, the general characteristics of the loan application process should be generated from a control-flow aspect. Secondly, it is necessary to identify the unfairness of business processes [10]. Since in the implementation of business processes there is a possibility of unfairness. Thirdly, research on fairness-aware process mining is limited, and we need to investigate other possible outcomes of unfairness in business processes besides [17]. Lastly, by analyzing the potential unfairness in business processes, it is necessary to generate general statements regarding fairness not only in the loan application process, but also other similar business processes. [8], [9].

Algorithm engineering development in fairness-aware business process. In developing integrated algorithm engineering, ontological, epistemological, and methodological dimensions have been stated [21]. Ontologically, the problems to be solved are how to deal with incomplete data in the loan application process, abnormal distribution, and how to give a fair assessment. Obtaining complete data containing sensitive variables of unfairness and integrating it with event logs is challenging. Therefore, it requires various approaches to obtain the appropriate data. The fairness algorithm will be developed and combined with the heuristic miner algorithm, which in its formation is influenced by the dependency graph. From an epistemological perspective, the purpose of developing this algorithm is to perform and assess the fairness of a business process.

4. Research Design

Our research method builds on methodological concepts of algorithm engineering and design science. Design science as described by Peffers et al [22] defines the overarching frame for this research. The design science research process consider six stages (Figure.1.)

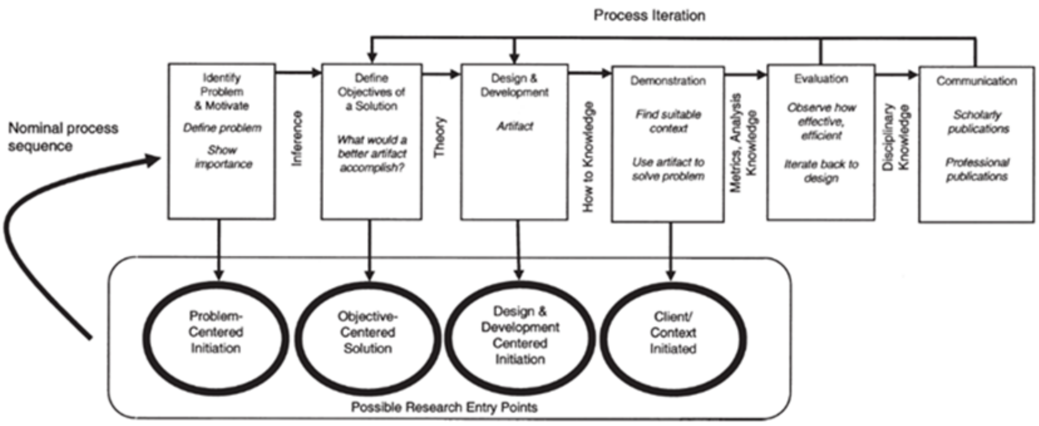

Figure 1: Design Research

The following is an explanation of each stage:

1. In the initial stage, the research problem is identified and motivated by conducting a thorough review of the existing literature on process mining and algorithmic fairness and its challenges. Fairness algorithms have been implemented in numerous case studies using machine learning. Based on the gaps and issues identified in previous studies, the main research question and sub-questions are formulated to guide the research.
2. Following problem identification, the objectives for a solution are defined. This involves specifying the criteria and metrics for evaluating fairness in the synthetic loan application process. Furthermore, the criteria and metrics can also be implemented in similar business processes.
3. In the third stage, the solution is designed and developed by proposing a framework for integrating heuristic miner algorithms and fairness measurement for the loan application process. The framework consists of a set of methods and techniques for assessing and enhancing fairness at different stages of the algorithmic pipeline.
4. In the fourth stage, the proposed solution is demonstrated and rigorously evaluated using synthetic loan application process data. On the other hand, focusing on model interpretability, the main approaches include in-process fairness mitigation and post-processing techniques. Nevertheless, these approaches have their own limitations. In-process fairness mitigation, with real-time implementation, may introduce operational complexities and delays, while post-processing techniques, may be more reactive than proactive, not effectively addressing initial biases in the lending context.
5. The fifth stage focuses on sharing the research results and insights with a broad audience. Communication and diffusion strategies may include publishing research papers in relevant journals and conferences.
6. In the final stage, a critical reflection and learning process is conducted. The strengths and limitations of the research are examined, and provide recommendations for future research directions.

5. Conclusion

The research proposes to address the gap by conducting an in-depth analysis of fairness concerns in process mining, developing new algorithms that explicitly incorporate fairness considerations, and evaluating their effectiveness. The research methodology aligns with design science and algorithm engineering principles, combining engineering research with empirical evaluations. This holistic approach seeks to contribute to the ethical enhancement of process mining techniques and their responsible application in real-world scenarios.

Acknowledgments

I would like to thank my thesis advisor, Jan Mendling, for his invaluable guidance and support in writing this proposal, and Stephan Fahrenkrog-Petersen for the helpful discussions about the research design and background.

References

[1] W. Van Der Aalst, W. van der Aalst, Data science in action, Springer, 2016.

[2] J. Munoz-Gama, N. Martin, C. Fernandez-Llatas, O. A. Johnson, M. Sepúlveda, E. Helm, V. Galvez-Yanjari, E. Rojas, A. Martinez-Millana, D. Aloini, et al., Process mining for healthcare: Characteristics and challenges, Journal of Biomedical Informatics 127 (2022) 103994.

[3] M. Jans, M. Eulerich, Process mining for financial auditing, in: Process Mining Handbook, Springer International Publishing Cham, 2022, pp. 445–467.

[4] G. Bernard, T. Boillat, C. Legner, P. Andritsos, When sales meet process mining: A scientific approach to sales process and performance management, in: Proceedings of the 37th International Conference on Information Systems (ICIS 2016), 2016.

[5] A. Bogarín, R. Cerezo, C. Romero, A survey on educational process mining, Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery 8 (2018) e1230.

[6] P. Ardimento, M. L. Bernardi, M. Cimitile, F. M. Maggi, Evaluating coding behavior in software development processes: A process mining approach, in: 2019 IEEE/ACM International Conference on Software and System Processes (ICSSP), IEEE, 2019, pp. 84–93.

[7] F. Rojos, F. Lucchini, J. M. González, D. Espinoza, J. Lee, J. P. S. Fernández, M. Arias, Process mining: Research in banking operations, BPI Challenge (2017).

[8] D. Dakic, D. Stefanovic, I. Cosic, T. Lolic, M. Medojevic, Business process mining application: A literature review., Annals of DAAAM & Proceedings 29 (2018).

[9] B. Kamala, B. Latha, Process mining and deep neural network approach for the prediction of business process outcome, in: 2022 International Conference on Communication, Computing and Internet of Things (IC3IoT), IEEE, 2022, pp. 1–4.

[10] D. Pessach, E. Shmueli, A review on fairness in machine learning, ACM Computing Surveys (CSUR) 55 (2022) 1–44.

[11] N. Kozodoi, J. Jacob, S. Lessmann, Fairness in credit scoring: Assessment, implementation and profit implications, European Journal of Operational Research 297 (2022) 1083–1094.

[12] W. Feng, Z. Wu, R. Gan, Y. Zheng, A study of fair prediction on credit assessment based on counterfactual fairness, in: 2022 International Conference on Machine Learning and Intelligent Systems Engineering (MLISE), IEEE, 2022, pp. 73–76.

[13] M. H. Teodorescu, X. Yao, Machine learning fairness is computationally difficult and algorithmically unsatisfactorily solved, in: 2021 IEEE High Performance Extreme Computing Conference (HPEC), IEEE, 2021, pp. 1–8.

[14] U. Gohar, S. Biswas, H. Rajan, Towards understanding fairness and its composition in ensemble machine learning, in: 2023 IEEE/ACM 45th International Conference on Software Engineering (ICSE), IEEE, 2023, pp. 1533–1545.

[15] M. S. Qafari, W. van der Aalst, Fairness-aware process mining, in: On the Move to Meaningful Internet Systems: OTM 2019 Conferences: Confederated International Conferences: CoopIS, ODBASE, C&TC 2019, Rhodes, Greece, October 21–25, 2019, Proceedings, Springer, 2019, pp. 182–192.

[16] T. Pohl, M. S. Qafari, W. M. van der Aalst, Discrimination-aware process mining: A discussion, in: International Conference on Process Mining, Springer, 2022, pp. 101–113.

[17] T. Pohl, A. Berti, M. S. Qafari, W. M. van der Aalst, A collection of simulated event logs for fairness assessment in process mining, arXiv preprint arXiv:2306.11453 (2023).
[18] N. Bhutta, A. Hizmo, D. Ringo, How much does racial bias affect mortgage lending? evidence from human and algorithmic credit decisions (2022).
[19] M. L. Kasmi, Machine Learning Fairness in Finance: An Application to Credit Scoring, Ph.D. thesis, Tilburg University, 2021.
[20] M. S. A. Lee, L. Floridi, Algorithmic fairness in mortgage lending: from absolute conditions to relational trade-offs, Minds and Machines 31 (2021) 165–191.
[21] J. Mendling, B. Depaire, H. Leopold, Theory and practice of algorithm engineering, arXiv preprint arXiv:2107.10675 (2021).
[22] K. Peffers, T. Tuunanen, M. A. Rothenberger, S. Chatterjee, A design science research methodology for information systems research, Journal of management information systems 24 (2007) 45–77.

Towards a Threat Modeling Language for Vessel Navigation and Port Call Optimization: harborLang

Simon Hacks

Stockholm University, Stockholm, Sweden

Abstract

This paper presents harborLang, a novel threat modeling language tailored for the maritime sector, built on the Meta Attack Language (MAL) framework. harborLang addresses the unique security challenges in maritime transport by enabling modeling and mitigation of potential threats. Through integrating specific maritime domain knowledge, harborLang empowers stakeholders to construct comprehensive threat models, enhancing decision-making and operational safety in seaports and vessel navigation.

Keywords

Maritime Threat Modeling, harborLang, Operational Security, Domain Specific Language

1. Introduction

Maritime transport is fundamental for the global economy, as it accounts for over 80% of the world's trade [1]. At the same time, total emissions of the world fleet increased by 4,7% between 2020 and 2021, harming the global aim for a carbon-neutral future [2]. One angle to reduce emissions within maritime transport is to address the "sail-fast-then-wait" (SFTW) syndrome, which shows vessels sailing at a predetermined speed to their destination port to find port resources not ready, forcing them to wait as most ports in the world serve ships on a first-come-first-served basis. Communicating terminal or port readiness early allows ships to adjust speed and save fuel for up to 23% of the voyage. Besides advances in Information Technology (IT), optimizing cargo transport within one system is unsolved [3]. In addition, communication between relevant stakeholders in the different port call phases is generally low and disorganized.

To address this, the MaritIme juSt in time optimiSatION (MISSION) project will develop a fully digitalized voyage and port call optimization system. It will enable collaboration among stakeholders, thus allowing the synchronization of ship schedules, optimizing ship operations, and port services to enhance operations efficiency and reduce fuel consumption. The developed system will comprise many components that interact and be used in a critical infrastructure. Therefore, ensuring the integrity of the system is of significant importance. One part of achieving this is using threat modeling and attack simulations using the Meta Attack Language (MAL) capabilities [4, 5].

EMISA'24: 14th International Workshop on Enterprise Modeling and Information Systems Architecture, March 13–14, 2024, Potsdam, Germany

simon.hacks@dsv.su.se (S. Hacks)

0000-0003-0478-9347 (S. Hacks)

CEUR Workshop Proceedings (CEUR-WS.org)

This work presents the first step towards a MAL domain-specific language (DSL) built for the described problem, the so-called *harborLang*. Following Hacks et al. [6] for making MAL DSLs and the Unified Process for Ontology building (UPON) lite [7], we present the identification of the basic concepts in the domain relevant to the MISSION project.

The rest of this work is structured as follows. In the background section, we briefly introduce MAL and present related works that serve as input for harborLang. Next, we give an overview of the research method used to develop harborLang throughout the project, followed by an outline of harborLang itself and the conclusions.

2. Background

2.1. The Meta Attack Language

A MAL DSL contains the main elements encountered on the domain under study, so-called `assets`. The assets contain `attack steps`, representing the possible attacks.

An `attack step` can be connected with the succeeding `attack steps` so that an attack path is created. Those attack paths comprise attack graphs facilitated in attack simulation. An `attack step` can be either OR or AND, respectively indicating that performing any individual parental `attack step` is required (OR) or performing all parental `attack steps` is required (AND) for the current step to be performed. Attack steps of type OR are defined by the symbol | while AND attack steps are defined by the symbol & before their names.

Furthermore, `defenses` do not allow connected `attack steps` to be performed if they have the value TRUE, which represents them as enabled. Finally, `probability distributions` can be assigned to `attack steps` to represent the effort needed to complete the related `attack step` or the probability of the `attack step` to be possible.

`Assets` have relations between them that are used to create a model. Those relations are called `associations` and are defined by the <- - and - -> symbols. When associations are specified, a name for the association and cardinalities for both assets should be defined. Inheritance between `assets` is also possible, and each child `asset` inherits all the `attack steps` of the parent `asset`. Additionally, the `assets` can be organized into `categories` for purely organization reasons.

Listing 1 presents a domain-agnostic example of a MAL DSL to ease understanding. In this example, four modeled assets can be seen together with the connections of attack steps from one asset to another. In the `Host` asset on line 6, the *connect* attack step is an OR attack step while *access* is an AND attack step. Then, the -> symbol denotes the connected next attack step.

For example, if an attacker performs *phish* on the `User`, it is possible to reach *obtain* on the associated `Password` and, as a result, finally perform *authenticate* on the associated `Host`. In lines 29 to 39, the `associations` between the assets are defined.

Listing 1: Exemplary MAL Code

```
1   category System {
2     asset Network {
3       | access
4         -> hosts.connect
5     }
6     asset Host {
7       | connect
8         -> access
9       | authenticate
10        -> access
11      | guessPwd
12        -> guessedPwd
13      | guessedPwd [Exp(0.02)]
14        -> authenticate
15      & access
```

```
  }
  asset User {
    | attemptPhishing
      -> phish
    | phish [Exp(0.1)]
      -> passwords.obtain
  }
  asset Password extends Data {
    | obtain
      -> host.authenticate
  }
}

associations {
  Network [networks] *
  <-- NetworkAccess -->
    * [hosts] Host
  Host [host] 1
    <-- Credentials -->
    * [passwords] Password
  User [user] 1
    <-- Credentials -->
    * [passwords] Password
}
```

2.2. Related Work

harborLang will mainly have two foundations. On the one hand, we base the language on existing MAL DSLs and situate it accordingly in the ecosystem of MAL DSLs [8] with concepts from the IT domain, such as classical office environments and with concepts from the Operational Technology (OT), such as the cyber-physical systems (CPS) controlling the vessels or the machinery in the harbor. coreLang [9] will cover the IT parts, which provide the basic concepts to model IT systems like applications, related hardware, and communication infrastructure. The OT parts will stem from icsLang [10], a language designed for industrial control systems (ICS).

On the other hand, we will rely on existing research in the maritime supply chain IT landscape describing the main components. For example, Port Community Systems (PCS) are electronic platforms that support communication and integration among various stakeholders within the port community. PCS focuses on improving service levels, partner networks, maritime and freight services, and fostering horizontal collaboration between seaport community partners [11, 12]. Another component is Terminal Operating Systems (TOS), a critical component of port and terminal operations, serving as the backbone for managing the day-to-day activities within a maritime terminal or port [13]. Moreover, there are systems to optimize the routes of the vessels, the (un)loading, and ensure the successful communication between the different parts [14].

3. Research Method

Hacks et al. [6] suggest following an Action Design Research (ADR) [15] approach to create MAL DSLs, and we follow this suggestion as harborLang will be developed in close exchange with the industrial partners of the MISSION project.

ADR structures research projects in four phases: (1) Problem formulation: The problem is defined by the project definition and will be further refined during the project. (2) Building, Intervention, and Evaluation: We will focus on the building aspect in this work, specifically identifying the core concepts in harborLang by applying UPON lite [7] as a lightweight method for ontology building. (3) Reflection and Learning: This will be ensured during the project by closely involving the industrial partners in developing harborLang utilizing workshops and interviews. (4) Formalization of Learning: The formalization results of the harborLang development will result in embedding it into the MAL DSL ecosystem [8] and identifying recurring patterns in the language development that will be used to improve future developments.

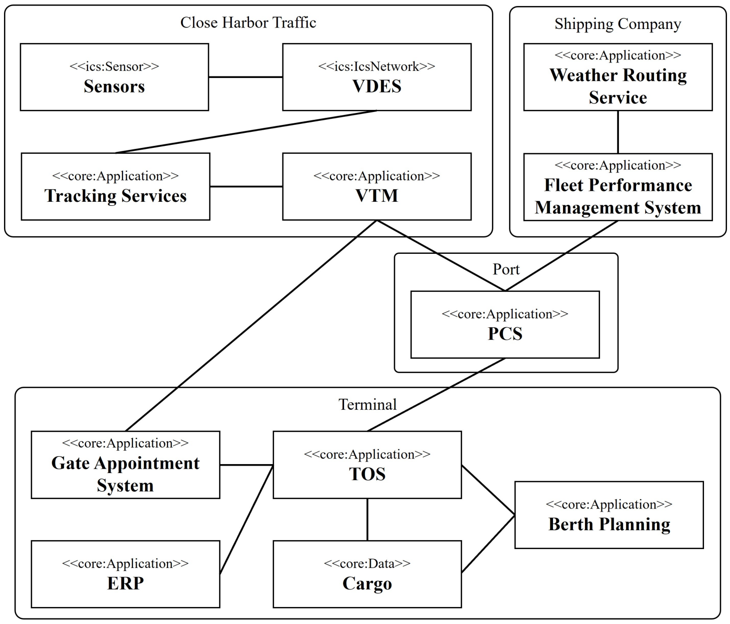

Figure 1: Main Components of Future harborLang

As we are focusing on the first step of building harborLang, we present next how we addressed the six steps of UPON lite [7] in a first iteration within this work: (1) Lexicon: In the first step, a lexicon of all terms in the domain is created including synonyms. To create this lexicon, we rely on existing research in the field [11, 12, 13, 14] and the partners' codified knowledge in the project proposal. (2) Glossary: The second step aims to unify the lexicon by identifying synonyms and providing a textual description of the single terms. Here, we restrict ourselves to identifying the synonyms in this iteration, as a detailed description of the different concepts is not yet needed. (3) Taxonomy: The third step focuses on defining a taxonomy of the terms within the glossary. Additionally to this activity, we identify possible hierarchies of harborLang concepts with coreLang [9] and icsLang [10]. The last steps, (4) Predication, (5) Parthood, and (6) Ontology, will not be further addressed in this iteration.

4. Outline of harborLang

The maritime supply chain IT landscape with seaports as intermodal hubs consists of various IT-systems [14] that support the different actors, e.g., **vessel traffic management (VTM)** service providers ensuring incoming/entering traffic based on **sensors** and **tracking services** communicating via very high frequency (VHF) data exchange system (**VDES**) [16]. Shipping companies use **fleet performance management systems** and **weather routing services** to provide the best routing by minimizing some metrics, e.g., fuel consumption or time.

Ports employ **port community systems (PCS)**. These are unique platforms that automate data, link individual existing systems of distinct stakeholders, and enable real-time data sharing for interaction and to reduce the administrative burden on ships [17].

Terminals use **terminal operating systems (TOS)** to plan and execute terminal operations, providing functionalities to control storage movement of **various cargo types** and planning of asset usage, labor, and equipment workload, i.e., **enterprise resource planning (ERP)**. **Berth planning** is frequently done using spreadsheet solutions and updating plans in a work-intense and time-consuming endeavor. **Gate appointment** systems help orchestrate port-bound cargo traffic. This leads to the first outline for harborLang represented in Figure 1.

5. Conclusions

This paper introduced the basic concepts of harborLang, a MAL DSL for maritime cybersecurity. By leveraging the Meta Attack Language (MAL), harborLang provides a robust framework for simulating and analyzing threats, thus contributing to safeguarding maritime operations.

As the global economy depends heavily on maritime transport, the role of threat modeling languages like harborLang becomes critical. We will continuously enhance harborLang in the MISSION project to integrate seamlessly with existing maritime IT infrastructures and its ability to evolve alongside emerging threats and technological advancements.

The challenges for the further development and implementation of harborLang involve addressing technological integration with existing maritime infrastructures, ensuring data privacy and security in maritime operations, and fostering effective collaboration among stakeholders. These challenges highlight the need for a scalable, flexible solution that adapts to the evolving technological landscape. Based on experience from previous projects, identifying concrete assets used in the infrastructure is challenging as concerning information is often classified. Moreover, the amount of involved organizations is large, leading to different abstraction levels of the provided information.

As harborLang continues establishing itself as a framework in maritime threat modeling, several avenues for future work emerge, promising to enhance its capabilities and applicability: Firstly, one can develop capabilities within harborLang to identify and model threats and recommend or automate response actions. Secondly, one could engage with global entities such as the International Maritime Organization (IMO) to standardize threat modeling protocols and ensure that harborLang aligns with international regulations and best practices.

Acknowledgments

This work has received funding from European Union's HORIZON research and innovation programme under the Grant Agreement no. 101138583.

References

[1] United Nations, Review of maritime transport 2023, 2023.
[2] T. Hák, S. Janoušková, B. Moldan, Sustainable development goals: A need for relevant indicators, Ecological indicators 60 (2016) 565–573.

[3] I. Sung, H. Zografakis, P. Nielsen, Multi-lateral ocean voyage optimization for cargo vessels as a decarbonization method, Transportation Research Part D: Transport and Environment 110 (2022) 103407.
[4] P. Johnson, R. Lagerström, M. Ekstedt, A meta language for threat modeling and attack simulations, in: Proceedings of the 13th International Conference on Availability, Reliability and Security, ACM, 2018, p. 38.
[5] W. Wideł, S. Hacks, M. Ekstedt, P. Johnson, R. Lagerström, The meta attack language-a formal description, Computers & Security 130 (2023) 103284.
[6] S. Hacks, S. Katsikeas, E. Rencelj Ling, W. Xiong, J. Pfeiffer, A. Wortmann, Towards a systematic method for developing meta attack language instances, in: International Conference on Business Process Modeling, Development and Support, Springer, 2022, pp. 139–154.
[7] A. De Nicola, M. Missikoff, A lightweight methodology for rapid ontology engineering, Commun. ACM 59 (2016) 79–86. doi:10.1145/2818359.
[8] S. Hacks, S. Katsikeas, Towards an ecosystem of domain specific languages for threat modeling, in: International Conference on Advanced Information Systems Engineering, Springer, 2021, pp. 3–18.
[9] S. Katsikeas, S. Hacks, P. Johnson, M. Ekstedt, R. Lagerström, J. Jacobsson, M. Wällstedt, P. Eliasson, An attack simulation language for the it domain, in: H. Eades III, O. Gadyatskaya (Eds.), Graphical Models for Security, Springer International Publishing, Cham, 2020, pp. 67–86.
[10] S. Hacks, S. Katsikeas, E. Ling, R. Lagerström, M. Ekstedt, powerlang: a probabilistic attack simulation language for the power domain, Energy Informatics 3 (2020).
[11] V. Caldeirinha, J. L. Nabais, C. Pinto, Port community systems: accelerating the transition of seaports toward the physical internet—the portuguese case, Journal of Marine Science and Engineering 10 (2022) 152.
[12] A. Moros-Daza, R. Amaya-Mier, C. Paternina-Arboleda, Port community systems: A structured literature review, Transportation Research Part A: Policy and Practice 133 (2020) 27–46.
[13] M. Grafelmann, C. Zlotos, A.-K. Lange, C. Jahn, Modelling the it and business process landscapes at inland intermodal terminals, in: Data Science in Maritime and City Logistics: Data-driven Solutions for Logistics and Sustainability. Proceedings of the Hamburg International Conference of Logistics (HICL), Vol. 30, Berlin: epubli GmbH, 2020, pp. 159–179.
[14] P. Pagano, S. Antonelli, A. Tardo, C-ports: A proposal for a comprehensive standardization and implementation plan of digital services offered by the "port of the future", Computers in Industry 134 (2022) 103556.
[15] M. K. Sein, O. Henfridsson, S. Purao, M. Rossi, R. Lindgren, Action design research, MIS quarterly (2011) 37–56.
[16] J. Šafár, C. Hargreaves, N. Ward, The vhf data exchange system, in: Antennas, Propagation & RF Technology for Transport and Autonomous Platforms 2017, IET, 2017, pp. 1–8.
[17] C. J. Pinto, P. F. Anunciacao, European seaports information systems. the impacts of directive 2010/65/eu, Economics and Culture 17 (2020) 38–49.

Domain-Specific Conceptual Modeling: An instrument for increasing productivity in system development

Martin Paczona [1], Heinrich C. Mayr [2] und Guenter Prochart[3]

Keywords: domain-specific conceptual modeling, DSML, testbed model, evaluation, productivity, modeling tool

Abstract: This EMISA24 "Current Research Talk" is based on the authors' paper [PMP24], recently published in the journal Data and Knowledge Engineering. We address the question of whether and how domain-specific modeling can increase productivity in the development of technical systems in an industrial setting. For, managers' decisions are ultimately based on whether or not the use of a new method pays off. The research was intertwined with a project in which we collaborated with a company to develop a domain-specific modeling method and tool for designing and implementing testbeds for electric vehicles.

Summary

Our focus is on productivity in the process of technical system development, where productivity is roughly defined as the relationship between production output and production input such as labor, capital, and environment. The research questions addressed are:

- R1: What are the productivity factors in technical system development?
- R2: How are these factors affected by using a DSMM for development?
- R3: What effort should one expect to expend in developing a DSMM?

We tackled these questions in accompanying research for a project in which we developed a DSMM for Electric Vehicle Testbeds (EVT) [PM19; PMP20] in cooperation with an Austrian enterprise. In the paper we (1) introduce a taxonomy of productivity factors in system development, (2) examine how these factors are influenced with domain-specific modeling, and (3) apply them to the development of our method itself.

This is a work in the field of Design Science. We have (1, "*Rigor Cycle*") reviewed the existing knowledge in the research area, (2, "*Relevance Cycle*") developed an artifact and (3, "*Evaluation Cycle*") assessed our DSMM in case studies addressing the following aspects: (1) Appropriateness of the DSML, (2) Tool introduction and first exercises, (3) Remodeling graphical testbed drawings from previous projects, (4) Generating models from customer orders, (5) Applying the DSMM in real projects, (6) Development phase, and (7) Usage in support.

1 Alpen-Adria-Universität Klagenfurt , 9020 Klagenfurt , Austria, mapaczona@edu.aau.at, https://orcid.org/0000-0002-6330-5974

2 Alpen-Adria-Universität Klagenfurt , 9020 Klagenfurt , Austria, heinrich.mayr@aau.at, https://orcid.org/0000-0001-5770-8091

3 Virtual Vehicle Research GmbH , 8010 Graz , Austria, guenter.prochart@v2c2.at

The productivity drivers in development processes, that we have identified, are (together with the results gained from the evaluation) the following:

1. **Error prevention and correctness support**: Errors in the early development phases lead to massive costs if they are not prevented or detected in time. DSMMs provide tool-supported use of domain-specific conceptualizations, guidelines, style guides, templates, constraints etc. Evaluation: Over 50% of the analyzed testbeds had one or more constraint violations that could have been detected by the implemented constraint checks. I.e., our approach reduces the number of errors in testbed design.
2. **Methods and tools tailored to design & development**: Conceptualizations that correspond to the tasks and procedures in a domain can be learned and applied more easily. Modern DSMMs provide such conceptualizations as well as appropriate notations. Evaluation: For quantifying acceptance we analyzed the modeling effort, the number of delete actions during modeling and the understandability. It turned out that the DSML is easy to understand, and that development and modification effort is reduced. The analysis of the modeling protocols of inexperienced users showed that they proceeded very purposefully and only had to make a few corrections.
3. **Reuse of artifacts**: Models are artifacts; repositories for their reuse offer at least a search function. More advanced features support multiple metamodels and thus models formed in different languages, and their consistent composition or the reuse of templates. Evaluation: Our repository contains over 100 models, allows the import and export of models and other artifacts, e.g., parts lists. Template models can be reused for new testbed designs.
4. **Automation of work units**: In design and development this is possible wherever artifacts can be generated with the help of rules from specifications or models. The key tasks in testbed development (creating the software configuration, developing circuit diagrams and specifying of control parameters) were previously performed manually and are now handled in the EVT Modeler by generators implemented in AdoScript and Python. In the development, time-consuming checks are shortened by automatic constraint checking.
5. **Design for subsequent maintainability**: Every technical system must be maintained and adapted to new requirements. Thus, easy maintainability must be planned for in design and development. Adaptability might be supported by DSML extension mechanisms.
6. **Appropriate product support**: Comprehensive and easy-to-understand system descriptions can be a great help in product support. This is also true for DSMMs. Evaluation: Our instruments help the support staff to detect design errors and provides suggestions for improvement. Testbed-specific software configurations and control parameters can be re-generated automatically without the need for manual tuning.

Literatur

[PM19] Paczona, M.; Mayr, H. C.: Model-Driven Mechatronic System Development. In: Proc. 2019 IEEE 15th Int. Conf. on Automation Science and Engineering (CASE). IEEE, Vancouver, BC, Canada, S. 1730–1736, 2019.

[PMP20] Paczona, M.; Mayr, H. C.; Prochart, G.: Model-Based Generation of Software Configurations in Mechatr. Systems. In: Modellierung 2020. LNI P-302, 2020.

[PMP24] Paczona, M.; Mayr, H. C.; Prochart, G.: Increase development productivity by domain-specific conceptual modeling. Data & Knowledge Engin. 150/, 2024.

Investigating the impact of representation features on decision model comprehension (Extended Abstract)[5]

Djordje Djurica [1], Tyge F. Kummer [2], Jan Mendling [1,3], and Kathrin Figl [4]

Abstract: Decision models play a crucial role in the development of information systems for tasks such as system analysis and design, as well as compliance management. The effective presentation of these models is essential to ensure their accuracy and completeness. Existing research on their cognitive effectiveness remains inconclusive. Our study advances understanding by examining the detailed representation features of decision models, including type (tree versus table), structure (expanded versus frugal), and design (monochromatic versus colored). We demonstrate that the use of color can improve model-task fit, and that structural features can enhance comprehension. Utilizing eye-tracking, we analyzed the underlying mechanisms of these effects. Our findings provide valuable insights for cognitive information systems research and practical applications, offering guidance for both users and developers of decision models.

Keywords: cognitive fit theory, decision models, experiment, color highlighting

1 Introduction

Decision-making processes, including the data employed and the relevant policies and regulations, can be effectively and visually represented and organized using decision models. These models are designed across a broad range of domains, from compliance management [RG+18] to forensic data matching [Xu07]. Careful design and quality assurance are critical in creating decision models. Although research on decision modeling has been ongoing since the 1980s, studies focusing on the cognitive aspects often produce conflicting results and face challenges in reconciling these findings with established theoretical frameworks. Cognitive Fit Theory (CFT) is a fundamental theory in information systems (IS) research [Ve06; Ve91] and has been utilized as a reference theory in conceptual modeling research [Kh06]. A persistent open research question is how representations can be more finely characterized beyond the basic dichotomy of spatial versus symbolic. In our paper, we address this research question by extending CFT. More specifically, we revisit the debate over the relative strengths of different types of decision models, including decision tables and trees.

1 Vienna University of Economics and Business, Vienna, Austria, djordje.djurica@wu.ac.at, https://orcid.org/0000-0002-3656-8314
2 Queensland University of Technology (QUT), Brisbane, Australia, t.kummer@qut.edu.au, https://orcid.org/0000-0001-5272-5255
3 Humboldt-Universität zu Berlin, Berlin, Germany,
4 University of Innsbruck, Innsbruck, Austria, Kathrin.Figl@uibk.ac.at, https://orcid.org/0000-0002-0777-2295
5 The original article has been published in the European Journal of Information Systems (EJIS) [Dj23]

2 Experimental Study

In our study [Dj23], our objective was to identify the factors driving cognitive fit at a detailed level. To achieve this, we extended previous research that primarily compared types of representations by also considering the manner in which information is presented, contrasting fast-and-frugal decision models with the more conventional expanded ones. Furthermore, we investigated the impact of color highlighting as a key visual factor that can significantly improve the understanding of these models. To examine the effects of these elements, we test three hypotheses centered on the effectiveness of decision models grounded in cognitive analysis of visual search. The formulated hypotheses were tested through a controlled experiment complemented by an explorative eye-tracking study. The controlled experiment employed a within-subject design, incorporating three factors: representation type, representation structure, and representation design. Example decision tables can be seen in Figure 1. The experiment involved 32 comprehension questions. A total of 139 participants, recruited via Prolific, participated in our study. The results corroborate our hypotheses: decision tables lead to higher comprehension accuracy than decision trees; frugal representations are more effective than expanded ones; and using color enhances comprehension accuracy for decision trees, while it appears to decrease it for decision tables. Our eye-tracking study indicates that this decrease might be attributed to distraction.

a

High Income	Double Income Household	Prior Debt	Credit Score
TRUE	TRUE	TRUE	Good Score
TRUE	TRUE	FALSE	Good Score
TRUE	FALSE	TRUE	Bad Score
TRUE	FALSE	FALSE	Bad Score
FALSE	TRUE	TRUE	Bad Score
FALSE	TRUE	FALSE	Bad Score
FALSE	FALSE	TRUE	Bad Score
FALSE	FALSE	FALSE	Bad Score

b

High Income	Double Income Household	Prior Debt	Credit Score
TRUE	TRUE	TRUE	Good Score
TRUE	TRUE	FALSE	Good Score
TRUE	FALSE	TRUE	Bad Score
TRUE	FALSE	FALSE	Bad Score
FALSE	TRUE	TRUE	Bad Score
FALSE	TRUE	FALSE	Bad Score
FALSE	FALSE	TRUE	Bad Score
FALSE	FALSE	FALSE	Bad Score

c

High Income	Double Income Household	Prior Debt	Credit Score
FALSE	-	-	Bad Score
TRUE	TRUE	-	Good Score
TRUE	FALSE	TRUE	Bad Score
TRUE	FALSE	FALSE	Good Score

d

High Income	Double Income Household	Prior Debt	Credit Score
FALSE	-	-	Bad Score
TRUE	TRUE	-	Good Score
TRUE	FALSE	TRUE	Bad Score
TRUE	FALSE	FALSE	Good Score

Fig. 1: Examples of Decision Tables Stimuli (a) Monochromatic Expanded, (b) Colored Highlighted Expanded, (c) Monochromatic Frugal, (d) Colored Highlighted

3 Discussion & Conclusion

In our study, we discovered that the effectiveness of a decision model type depends on its interaction with other representational factors, such as the structure of the representation (expanded vs. frugal) and the design elements (monochromatic vs. colored). Our findings extend existing arguments of cognitive fit theory by demonstrating that using color can compensate for a mismatch between the decision model and the task at hand and that structural features can significantly enhance model comprehension. Furthermore, our results contribute to integrating cognitive fit theory with the research on fast and frugal trees and feature integration theory.

References

[Dj23] Djurica, D.; Kummer, T.-F.; Mendling, J.; Figl, K.: Investigating the impact of representation features on decision model comprehension. European Journal of Information Systems, pp. 1–25, 2023.

[Kh06] Khatri, V.; Vessey, I.; V. Ramesh, P. C.; Sung-Jin, P.: Understanding Conceptual Schemas: Exploring the Role of Application and IS Domain Knowledge. Information Systems Research 17/1, pp. 81–99, 2006.

[RG+18] Rosso, N.; Giabbanelli, P., et al.: Accurately inferring compliance to five major food guidelines through simplified surveys: applying data mining to the UK National Diet and Nutrition Survey. JMIR public health and surveillance 4/2, e9536, 2018.

[Ve06] Vessey, I.: The theory of cognitive fit: One aspect of a general theory of problem solving? In: Human-computer Interaction and Management Information Systems: Foundations. Routledge, pp. 141–183, 2006.

[Ve91] Vessey, I.: Cognitive fit: A theory-based analysis of the graphs versus tables literature. Decision Sciences 22/2, pp. 219–240, 1991.

[Xu07] Xu, J.; Wang, G. A.; Li, J.; Chau, M.: Complex problem solving: identity matching based on social contextual information. Journal of the Association for Information Systems 8/10, p. 31, 2007.

Process Mining Meets Visual Analytics: The Case of Conformance Checking [Extended Abstract][1]

Luise Pufahl[2], Michael Grohs, Lisa-Marie Klein, Jana-Rebecca Rehse[3]

Abstract:

This extended abstract summarizes a study on the visualization of conformance checking results in process mining, presented at HICSS 2023. Conformance checking compares intended and actual business process behaviors using IT system logs. Our study examined the visualization features of both academic and commercial process mining tools. We found these tools offer visualizations for quantifying conformance, breaking it down, localizing, and explaining deviations. However, there is a need for structured research on process analysts' visualization needs and the interaction between data, analysts, and visualizations.

Keywords: Process Mining; Conformance Checking; Visualization; Visual Analytics

Organizations execute a multitude of business processes to deliver products and services to their customers. If those processes are supported by IT systems, these systems can capture data about this execution in so-called event logs. With process mining [Aa12] techniques, organizations can analyze these event logs and gain insights into their processes in an evidenced-based fashion. It allows them to discover real-world process behavior, measure cycle times, and compare the executed to the intended process. This later task is called conformance checking. It is highly relevant for organizations [EAt19] because it allows to check whether the process execution follows specific rules defined by regulations or best practices. Research in conformance checking has focused on improving the algorithms' computational efficiency or extending them by data and organizational aspects of processes [Du19]. However, another relevant aspect is that when analyzing large amounts of data, the visualization of the results [Gs17] aids analysts in quickly understanding information, leading to more effective organizational decisions.

Our study analyzed the visualization capabilities of process mining tools for conformance checking, laying the groundwork for future research. Using a structured literature review, we identified nine academic tools and selected commercial ones referenced in [FA20].

[1] This article summarizes the problem, approach, and selected findings of a study published as Jana-Rebecca Rehse, Luise Pufahl, Michael Grohs, Lisa-Marie Klein: Process Mining Meets Visual Analytics: The Case of Conformance Checking, Proceedings of the 56th Hawaii International Conference on System Sciences (2023) `https://hdl.handle.net/10125/103299` [Re23]

[2] Technische Universität München, School of CIT, 76074 Heilbronn, Deutschland luise.pufahl@tum.de

[3] University of Mannheim, Management Analytics, 68161 Mannheim, Deutschland \{firstname.lastname\}@uni-mannheim.de

We analyzed these tools using a public event log, creating a process model with 20% noise filtering. We recorded how each tool presented conformance checking results. Then, we analyzed them with regards to *what* data was visualized (e.g., conformance levels or deviation distributions), *why* it was visualized (e.g., quantify conformance level or compare conformance distribution), and *how* the data was portrayed (e.g., line diagrams, or process diagrams), using the visualization idioms from the DataViz catalogue[4].

The results show that the presentation of conformance checking results can be classified into four high-level categories according to the *why* question: (1) quantify conformance, (2) break down and compare conformance, (3) localize and show deviations, and (4) explain and diagnose deviations. In the first category, we observed that academic and commercial tools provide different summarized conformance measures, such as fitness or conformance rates, mainly as numbers or colored numbers. An interaction by selection, e.g., the measure, is rarely possible. In the second category, only the commercial tools break down the summarized conformance measure along different dimensions, such as variants, case attributes, time, or location. Bar charts are mainly used, with an occasional scatter plots or bubble maps. In the third category, the tools visualize deviations in process diagrams, flowcharts, or chevrons. Different granularities are used, including the whole event log, specific variants, certain deviations, or deviation types. The user can usually not decide which kind of granularity they prefer. Finally, few tools provide a presentation for explaining and diagnosing a deviation by means of textual descriptions, numbers, and tables for the case attribute correlation. The paper closes with a future research discussion, where it observes that the users‘ information needs still have to be systematically analyzed, and the selection of suitable visualization idioms has to be evaluated.

References

[Aa12] van der Aalst, Wil et al.: Process Mining Manifesto. In: BPM Workshops. pp. 169–194, 2012.

[Du19] Dunzer, Sebastian; Stierle, Matthias; Matzner, Martin; Baier, Stephan: Conformance Checking: A State-of-the-Art Literature Review. In: Subject-Oriented BPM. 2019.

[EAt19] Emamjome, Fahame; Andrews, Robert; ter Hofstede, Arthur: A Case Study Lens on Process Mining in Practice. In: OTM. pp. 127–145, 2019.

[FA20] FAU: , Process Mining Software Comparison, 2020.

[Gs17] Gschwandtner, Theresia: Visual Analytics Meets Process Mining: Challenges and Opportunities. In: Data-Driven Process Discovery and Analysis. 2017.

[Re23] Rehse, Jana-Rebecca; Pufahl, Luise; Grohs, Michael; Klein, Lisa-Marie: Process Mining Meets Visual Analytics: The Case of Conformance Checking. In (Bui, Tung X., ed.): 56th Hawaii International Conference on System Sciences, HICSS 2023, Maui, Hawaii, USA, January 3-6, 2023. ScholarSpace, pp. 5452–5461, 2023.

[4] https://datavizcatalogue.com/

Describing Behavior Sequences of Fattening Pigs Using Process Mining on Video Data and Automated Pig Behavior Recognition (Extended Abstract)[1]

Arvid Lepsien[2], Andreas Melfsen[3], Jan Bosselmann[4], Agnes Koschmider[5], Eberhard Hartung[6]

Keywords: Process Mining; Unstructured Data; Behavior Sequences; Fattening Pigs

1 Motivation

Process mining is a well-established technique for gaining insights into event data. It allows significant insights into event data in terms of identifying process anomalies, giving hints between as-is and to-be process states or making predictions based on data. Although process mining has been successfully applied in many application domains like healthcare, finance, and manufacturing, additional domains might also benefit from process mining like life and natural sciences. However, these domains mainly do not rely on structured business data that is expected as input for process mining algorithms. Rather, data from these domains first has to be efficiently pre-processed.

This paper suggests process mining as an approach to identify behavioral patterns of fattening pigs from video data. The goal of this approach is to demonstrate that process mining might be a valuable tool for understanding the behavior of pigs by considering and analyzing their behavior sequences. Furthermore, additional insights can be gained in terms of temporal and spatial analysis about the division of the pig pen in functional areas. In this way, new implications might be found about pig behavior compared to existing state-of-the art approaches in the field.

[1] This extended abstract summarizes our original research article by the same title which appeared in *Agriculture 13 (8)* [Me23]. The project ProcessPig is funded by the European Union within the framework of the European Innovation Partnership (EIP-AGRI) and the state progam rural areas of the state Schleswig-Holstein (LPLR) for three years (www.eip-agrar-sh.de).

[2] Kiel University, Group Process Analytics, Kiel, Germany, ale@informatik.uni-kiel.de

[3] Kiel University, Institute of Agricultural Engineering, Faculty of Agricultural and Nutritional Sciences, Kiel, Germany, amelfsen@ilv.uni-kiel.de

[4] Kiel University, Group Process Analytics, Kiel, Germany

[5] University of Bayreuth, Business Informatics and Process Analytics, Fraunhofer FIT, Bayreuth, Germany, agnes.koschmider@uni-bayreuth.de

[6] Kiel University, Institute of Agricultural Engineering, Faculty of Agricultural and Nutritional Sciences, Kiel, Germany

2 Implementation

The implementation is based on the approach presented in [LKK23], which was adapted for the use case of behavior sequencing of fattening pigs. Particularly, video sequences recorded by a camera observing a pig pen are prepared for process mining by: (1) pre-processing of the raw videos, (2) object recognition and tracking, (3) activity recognition, (4) event abstraction, (5) case correlation, and (6) trace clustering. Following these steps results in an event log, which can be subsequently used for process mining. Each of these steps has been implemented accordingly.
Process discovery was applied to extract models of various behavioral patterns. These models can then be used to monitor the occurrence and frequency of different behavioral patterns and detect behavioral anomalies. For example, one model that was extracted describes the typical sequence of activities associated with soiling behavior, enabling the monitoring of the context associated with this behavior and the detection of outliers. Quantitative evaluation of the performance of the object detection, object tracking, and activity recognition models show high accuracy indicating the usefulness of the results. The quality of the discovered process models was evaluated by domain experts, who confirmed that the models describe common behavioral patterns known from pig behavioral research.

3 Conclusion

Our approach allows to identify individual behavioral patterns and to cluster them into unique sequences, which supports efficient behavioral pattern recognition. The limitations of the current implementation are that (1) the used activity recognition model requires the activity classes to be defined a priori, meaning that only activities that are known to be part of the targeted behavioral patterns can be detected, and (2) cases are extracted based on predefined start and end activities. Future analysis could unveil additional behavioral sequences with different start and endpoints. Overall, our approach promises an automatic assessment of pig behavior and behavioral sequences based on video data. In this way, our approach lays the foundation for informed decisions for different stakeholders (i.e., transparency of the pig treatment). In the future, we plan to further adapt our approach for multi-area pens with or without freely ventilated conditions. This corresponds to realistic settings in additional pig pens that might also benefit from our approach.

References

[LKK23] Lepsien, A.; Koschmider, A.; Kratsch, W.: Analytics Pipeline for Process Mining on Video Data. In: BPM 2023 Forum. Lecture Notes in Business Information Processing, Springer, Cham, pp. 196–213, 2023.

[Me23] Melfsen, A.; Lepsien, A.; Bosselmann, J.; Koschmider, A.; Hartung, E.: Describing Behavior Sequences of Fattening Pigs Using Process Mining on Video Data and Automated Pig Behavior Recognition. Agriculture 13/8, p. 1639, 2023.

Addressing the Log Representativeness Problem using Species Discovery (Extended Abstract)[1]

Martin Kabierski, Markus Richter, Matthias Weidlich[2]

1 Motivation

Event logs are generated during the enactment of process-centric information systems and form the basis for optimization, monitoring, and enhancement initiatives of said systems [vdA16]. As such, they enable a data-driven and unbiased evaluation of the as-is state of the underlying processes. Yet, since at any time, event logs represent merely a sample of the whole possible behaviour of the information system, insights are only actionable should the event log be *representative* of the information system from which it is derived. Therefore, the question arises of how the representativeness of an event log L with respect to its generative system P can be quantified, given that only L is present. In this work, we argue, that *representativeness* of an event log needs to be assessed with an intended analysis question in mind and discuss *log completeness* as one important facet of representativeness. We show how established estimators from biodiversity research can be utilized to quantify log completeness.

2 Log Completeness-based Representativeness

Roughly speaking, representativeness implies, that a sample L reflects the relevant properties of its corresponding population P. Insights about P derived from an event log L can only be considered reliable, if it has been verified, that log L is representative of P for the intended analysis purpose. One aspect of *representativeness*, among others, is *log completeness*, i.e. how many of the distinct potentially generated values of P are actually manifested in L. Quantifying the completeness of an event log allows the identification of incomplete log dimensions and the assessment of whether the event log contains "enough" information, to properly represent the diversity and complexity of system P. As such, completeness measures serve as a quantification of event log quality and the reliability of derived insights.

[1] The original article was presented at the 5th International Conference on Process Mining (2023) in Rome, Italy [KRW23]

[2] All authors are with the Humboldt-Universität zu Berlin, Department of Computer Science, Berlin, Germany, {martin.kabierski | markus.richter | matthias.weidlich}@hu-berlin.de

3 Biodiversity Measures as Completeness Measures

Let there be a generative system P with S_P being the unknown number of distinct values in a relevant dimension, each of which we call a *species* of the dimension. Let $L \subseteq P$ be an event log with observed number of species $S_{obs} \leq S_P$. Based on these notions, a completeness measure of L shall relate S_{obs} to the unknown S_P. This problem is equivalent to the question of estimating the species richness of an ecosystem in biodiversity research [Co09], where based on a sample of individuals of different species, the total number of expected species in the ecosystem is to be estimated. For this, established species richness estimators, such as the parameter-free Chao2-estimator [Ch87] can be employed. Based on this estimate, different completeness measures can be derived:

- completeness: the fraction of species from P, that are contained in L
- coverage: the fraction of total probability space the observed species cover
- the number of additional traces needed, to reach a target completeness g

These derived metrics enable the assessment of L regarding sufficient completeness, the importance of missing values in terms of occurrence probabilities and expected additional data collection efforts needed to ensure a more complete, and thus, a more representative, event log.

4 Evaluation

We calculated the proposed log completeness measures for four publicly available event logs. Our results show, that none of the logs can be considered as complete in any but the most simple log dimensions, and that, depending on the considered log dimension, a significant number of additional traces would be needed to reach proper completeness. Further experiments on systematically partitioned event logs reveal, that the completeness of different subprocesses differs significantly.

Bibliography

[Ch87] Chao, Anne: Estimating the population size for capture-recapture data with unequal catchability. Biometrics, S. 783–791, 1987.

[Co09] Colwell, Robert K et al.: Biodiversity: concepts, patterns, and measurement. The Princeton guide to ecology, 663:257–263, 2009.

[KRW23] Kabierski, Martin; Richter, Markus; Weidlich, Matthias: Addressing the Log Representativeness Problem using Species Discovery. In: 2023 5th International Conference on Process Mining (ICPM). IEEE, S. 65–72, 2023.

[vdA16] van der Aalst, Wil M. P.: Process Mining - Data Science in Action, Second Edition. Springer, 2016.