Quality Metrics for Business Process Event Logs Based on High Frequency Traces

Mohsen Mohammadi
Assistant Professor; Computer Department; Esfarayen University of Technology; Esfarayen, Iran Email: Mohsen@esfarayen.ac.ir

Iranian Journal of Information Processing and Management

Iranian Research Institute for Information Science and Technology (IranDoc)
ISSN 2251-8223
eISSN 2251-8231
Indexed by SCOPUS, ISC, & LISTA
Special Issue | Winter 2025 | pp. 143-166
Exploring the Relationship Between Data Quality and Business Process Management
https://doi.org/10.22034/jipm.2023.2013562.1405

Received: 14, Oct. 2023 **Accepted: 27, Dec. 2023**

Abstract: In today's data-centric business landscape, characterized by the omnipresence of advanced Business Intelligence and Data Science technologies, the practice of Process Mining takes center stage in Business Process Management. This study addresses the critical challenge of ensuring the quality of event logs, which serve as the foundational data source for Process Mining. Event logs, derived from interactions among process participants and information systems, offer profound insights into the authentic behavior of business processes, reflecting the organizational rules, procedures, norms, and culture. However, the quality of these event logs is often compromised by interactions among various actors and systems. In response, our research introduces a systematic approach that leverages Python and the pm4py library for data analysis. We employ trace filtering techniques and utilize Petri nets for process model representation. This paper proposes a methodology demonstrating a significant improvement in the quality metrics of extracted subprocesses through trace filtering. Comparative analyses between the original logs and filtered logs show enhancements in fitness, precision, generalization, and simplicity, highlighting the practical importance of trace filtering in refining complex process models. These findings offer practical insights for practitioners and researchers involved in process mining and modeling, highlighting the significance of data quality in obtaining precise and dependable business process insights.

Keywords: Process Quality, Quality Metrics, Business Process Model, Event Log

1. Introduction

In today's business environment, the essence of modern organizations lies in data. Advancements in business intelligence and data science technologies have provided organizations with the tools to enhance their data-driven capabilities and uncover hidden insights within vast databases. In the realm of Business Process Management (BPM), Process Mining stands out as a specialized variant of data-centric process analysis. Process mining leverages historical process data, typically extracted from diverse IT systems in the form of event logs. These event logs contain a wealth of information regarding the sequential activities, timestamps, and participants involved in business processes. Through the analysis of this data, process mining provides a nuanced understanding of how processes truly unfold within an organization, offering valuable insights into efficiency, bottlenecks, and potential areas for improvement (van der Aalst, 2016; Goel et al., 2022).

Event logs, typically considered the initial step in a process mining project, document the interactions among different elements such as process participants, automation components, data managers, and information systems. These interactions are influenced by organizational rules, procedures, norms, and culture. Utilizing event logs as the primary data source for Process Mining provides organizations with valuable insights into process performance, adherence to predefined process models, and opportunities for process improvement. This perspective on event logs suggests that quality issues observed in these logs are a result of interactions among various actors (such as process participants, automation, and data management), systems, and the overall context. Therefore, by delving deeper into the underlying dynamics, organizations can uncover valuable insights to enhance their process mining projects (Suriadi et al., 2017; Fischer et al., 2020; Andrews et al., 2022).

However, transitioning from discussing data quality challenges in event logs to the complexity of process models is crucial for maintaining a seamless flow. Many researchers (Suriadi et al., 2017; Andrews et al., 2019; Dakic et al., 2023) have demonstrated that actual event logs often exhibit a range of data quality challenges, including issues such as missing and imprecise attribute values. For instance, event logs may contain timestamps that are either inaccurate or defined with varying levels of precision. Despite the paramount significance of data quality,

the majority of process mining algorithms do not incorporate considerations for data quality or any data preprocessing details. This poses the potential hazard of yielding counterintuitive or even deceptive results (Goel et al., 2022). As the discussion shifts to process modeling, it becomes evident that addressing these data quality challenges is integral to ensuring the accuracy and reliability of the subsequent analysis, especially considering the intricacies of constructing and comprehending complex process models.

Process modeling plays a pivotal role in all Business Process Management (BPM) strategies. The most effective way to facilitate communication in process enhancement initiatives is through the utilization of process models (Reijers et al., 2015). The organizational complexity increases with a higher volume of input and output interactions among various departments (Krogstie, 2016). When faced with a complex process model, which is essentially a model that is challenging to decipher and lacks the clarity necessary for users to comprehend, traditional process discovery techniques often fail to provide adequate understandable information. In such instances, the complexity of the process model can severely impede its quality and hinder the retrieval of behavioral insights (De San Pedro et al., 2015; Fahland and Van Der Aalst, 2011). Various approaches have been suggested to address this issue, including the simplification of already mined models (De San Pedro et al., 2015) and the search for simpler structures within the logs (Leemans and Van Der Aalst, 2015; Tax et al., 2016). Despite the improvements these techniques offer in terms of process model comprehensibility, the structural complexity of real-world processes often persists, posing difficulties for users seeking to grasp them (Chapela-Campa et al., 2019). Therefore, this study aims to meticulously examine the quality metrics of extracted subprocesses, with a particular emphasis on utilizing high-frequency traces from event logs. This targeted approach shapes the research objectives, facilitating a more nuanced exploration of the intricate relationship between trace frequency and model precision.

The paper employs a systematic methodology to investigate the quality metrics of extracted subprocesses in real event logs. The rest of the paper is structured as follows: The following section introduces the background information and reviews some related works. In Section 3, we provide a description of the methodology

employed in this study. Section 4 contains the findings and discussion. Finally, the paper summarizes the conclusions in Section 5.

2. Literture review

This section provides insights into the evolution of event log evaluation methodologies and introduces cutting-edge techniques, paving the way for an in-depth examination of trace variants and filtering methods. It bridges the historical context of event log assessment to contemporary advancements, laying the groundwork for the subsequent methodology exploration. The following sub-sections introduce the background information, which is divided into two parts – the definition of an event log and the quality metrics associated with it, which will be the focus of our analysis; and an overview of relevant prior research.

2-1. Background

In the data-centric business landscape, (Hasanzadeh et al., 2012) propose a model for SOA governance maturity, complementing (Salehi et al., 2023) methodology for enhancing event log quality in Process Mining. (AliAbadi and Mohammadi, 2022) contribute insights into enterprise data integration using web services, collectively highlighting the importance of structured approaches in managing complex systems and ensuring data quality for accurate business process insights.

Event log:

As mentioned earlier, the initial step in process mining involves acquiring an event log, which can originate from an actual information system log or be constructed from historical data stored in a database. Regardless of its source, this event log must adhere to a specific structure. To enable subsequent analysis using process mining techniques, events of interest, which represent user actions in process instances, need to be recorded. When a substantial number of such events are collected, they form an event log. For effective process mining, each event within this log should include essential details such as a case ID (identifying the process instance), task name (identifying the activity), user name (identifying the participant), and timestamp (indicating when the task was completed). An ideal event log for process mining is specific to a single process and is organized as

a set of cases or traces in a multiple-record-case format, as illustrated in Figure 1. The essential information contained in each event, such as case ID, task name, and user name, ensures the log's suitability for thorough process analysis (Ferreira, 2017).

Case id	Task	User	Timestamp
1	*a*	u_1	2016-04-09 17:36:47
1	*b*	u_3	2016-04-11 09:11:13
1	*d*	u_6	2016-04-12 10:00:12
1	*e*	u_7	2016-04-12 18:21:32
1	*f*	u_8	2016-04-13 13:27:41
2	*a*	u_2	2016-04-14 08:56:09
2	*b*	u_3	2016-04-14 09:36:02
2	*d*	u_5	2016-04-15 10:16:40
1	*g*	u_6	2016-04-18 19:14:14
2	*g*	u_6	2016-04-19 15:39:15
1	*h*	u_2	2016-04-19 16:48:16
2	*e*	u_7	2016-04-20 14:39:45
2	*f*	u_8	2016-04-22 09:16:16
3	*a*	u_2	2016-04-25 08:39:24

Figure 1. a sample of event log (Ferreira, 2017)

In an event log, each case consists of the sequence of events carried out in a single execution of a process instance. Each unique sequence of events from the beginning to the end of a process instance is referred to as a variant. Each case or trace belongs to exactly one variant, and a variant may encompass one or more cases or traces (Suriadi, 2017).

The acquisition of an event log, fundamental to process mining, is detailed. Emphasizing a specific structure, the event log captures user actions with essential details like case ID, task name, user name, and timestamp. These logs, organized into cases or traces, provide a foundation for thorough process analysis.

Quality metrics:

To evaluate how effectively a process model captures observed behavior, we assess four distinct quality dimensions, as illustrated in Figure 2. Each dimension addresses a specific aspect of process model quality: simplicity, replay fitness, precision, and generalization. Simplicity refers to how easily the model can be understood by humans and is not related to the observed behavior. Given that different process models can describe the same behavior in various ways, opting for the simplest model is the preferred approach (Van der Aalst et al., 2012; Van der Aalst, 2016).

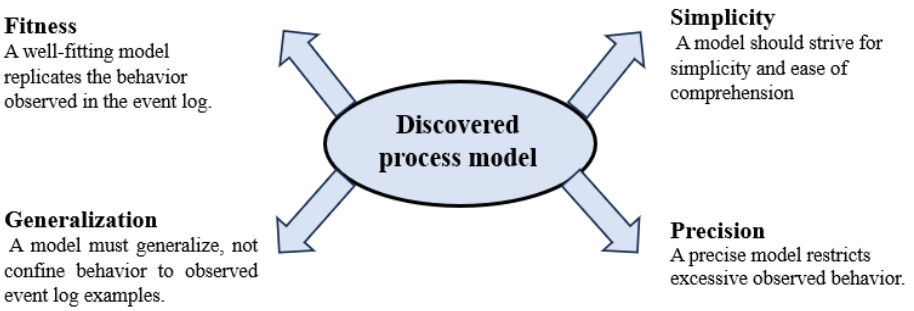

Figure 2. Quality dimensions for discovered process model (Buijs et al., 2014; Van Der Aalst, 2016; Janssenswillen et al., 2017)

Replay fitness, another quality dimension, measures the portion of the behavior documented in the event log that can be accurately replicated by the process model. Precision, on the other hand, measures the degree of behavior allowed by the process model but not actually observed in the event log. Both replay fitness and precision evaluate the alignment between the event log and the process model. However, it is crucial to note that the event log only captures a fraction of the potential behavior permitted by the system. Therefore, the dimension of generalization assesses whether the process model avoids becoming overly tailored to the observed event log behavior and accurately represents the broader system. In essence, generalization also reflects the model's ability to describe behavior that has not yet been observed within the system (Van Der Aalst, 2013; Buijs et al., 2014).

Quality metrics, including simplicity, replay fitness, precision, and generalization, are introduced as critical dimensions for evaluating the effectiveness of process

models. These metrics ensure an insightful analysis of observed behavior, guiding the subsequent investigation.

2-2. Related works

The Process Mining Manifesto (Van Der Aalst et al., 2012), introduces a maturity grading system ranging from 1-star to 5-star to assess the readiness of event logs for process mining analysis. According to the manifesto, event logs bestowed with 3, 4, or 5 stars are considered suitable for process mining analysis, while those with 1 or 2 stars are likely unsuitable. A 5-star-graded log, symbolizing excellent quality, trustworthiness, and completeness, is characterized by its automatic, systematic, and reliable recording, encompassing all events and their attributes with well-defined semantics. Conversely, 1-star-graded logs, signifying poor quality, often feature events that do not align with reality and may be incomplete, typically stemming from manual data recording.

(Mohammadi, 2017) conducts a thorough analysis of literature to identify and examine the diverse factors that impact the quality of business process models. It covers aspects such as modeling techniques, stakeholder involvement, data quality, and organizational context.

(Goel et al., 2022) emphasizes the practical enhancement of systematically evaluating event log quality, introduced by (Janssenswillen et al., 2017), facilitated through the utilization of the open-source R-package DaQAPO. Furthermore, (Andrews et al., 2018) have introduced the groundwork for the log query language QUELI, designed for the purpose of querying event logs to identify imperfections. Another noteworthy development is the introduction of RDB2Log, which enables the assessment of event logs during their creation from relational databases (Andrews et al., 2020). More recently, (Fischer et al., 2020) have put forward a framework designed to identify issues related to timestamps and generate a series of quality metrics, including measures of granularity and precision.

A “trace” represents a variant of an executed process, indicating that a single event log can contain numerous different trace variants. Trace clustering serves as a highly effective method for identifying noisy or abnormal traces and uncovering specific patterns of imperfections within event logs (Dakic et al., 2023). The primary purpose of trace clustering is to address issues related to the volume, complexity,

and granularity of event logs (Boltenhagen et al., 2019). Some researchers have applied trace clustering to identify similarities between trace variants, including incomplete traces. This enables the prediction of missing activity labels based on the succession relation matrix, as demonstrated by Liu et al. (2021). Additionally, trace clustering often serves as an initial step in the application of more advanced preprocessing techniques, such as those based on statistical inference, with the goal of reducing the complexity of an event log as explored by Ceravolo et al. in 2017. On the other hand, trace filtering techniques fall within the realm of event data transformation methods. They assess the likelihood of trace occurrence and eliminate events with lower frequencies of occurrence as discussed by Marin-Castro (2021).

Ireddy and Sergey (2023) provide a comprehensive and experimental perspective on existing literature, identifying trends, patterns, and gaps in the evaluation of process model quality. The authors utilize rigorous methodologies to quantitatively assess aggregated data, providing valuable insights for researchers and practitioners involved in process modeling. The paper not only consolidates current knowledge but also suggests potential avenues for future research, contributing to a deeper understanding of quality assessment in process modeling and informing best practices in this evolving field.

The mentioned works provide valuable practical insights for enhancing the effectiveness and reliability of process mining and modeling in real-world scenarios. The Process Mining Manifesto's maturity grading system offers a systematic approach to evaluate the appropriateness of event logs for process mining analysis, guaranteeing the quality and comprehensiveness of the underlying data. Mohammadi's exploration of factors influencing business process model quality provides practical considerations, including modeling techniques and stakeholder involvement, crucial for creating effective process models. Goel et al.'s tools, DaQAPO, QUELI, and RDB2Log, enhance the systematic evaluation of event log quality, providing practical solutions for researchers and practitioners. Fischer et al.'s framework addresses timestamp-related challenges in event log data, contributing practical measures for enhancing quality metrics. Trace clustering and filtering techniques, as discussed by Dakic et al. and others, offer practical methods to handle the complexity of event logs and enhance the overall quality of

process models. Ireddy and Sergey's comprehensive review not only consolidates current knowledge but also provides valuable guidance for future research. It serves as a practical resource for individuals involved in process modeling, offering insights to enhance quality assessment and inform best practices in this dynamic field.

3. Method

As it mentioned earlier, Process mining is a data-driven methodology that leverages event data logs to discover, analyze, and improve business processes. It provides insights into how processes are executed in reality, offering a visual representation of workflows, dependencies, and performance metrics. By examining event logs generated by information systems, process mining aims to enhance transparency, identify bottlenecks, and optimize business processes (Van Der Aalst, 2016). In this section, we provide a detailed overview of the steps comprising the proposed method, as illustrated in Figure 3. The methodology relies on key tools and techniques, notably Python and pm4py, which is a Python library encompassing a wide array of process mining algorithms. Two real event logs are adopted form IEEE Task Force on Process Mining (https://community.data.4tu.nl) including Loan application process of a Dutch financial institute (BPI Challenge 2017) and Purchase order handling process (BPI Challenge 2019).

The proposed methodology relies on key tools, particularly Python, and the pm4py library. Python serves as a versatile and widely-used programming language, offering a rich ecosystem of libraries suitable for data manipulation, analysis, and visualization. Pm4py, a specialized Python library, encompasses a diverse set of process mining algorithms, providing a comprehensive toolkit for handling event logs, extracting valuable insights, and constructing process models (Berti et al., 2019).

Petri nets were chosen as the representation for the discovered process models due to their unique advantages in the context of process mining. Petri nets offer a concise and unambiguous way to represent the behavior of processes, facilitating clear visualization and analysis. Their ability to encapsulate the behavior of an accepting Petri net, covering all traces from the initial marking to the final marking, makes them well-suited for capturing the flow and dependencies

within a business process. Furthermore, Petri nets are widely adopted in both BPM and Workflow Management (WFM) systems, providing a standardized and widely recognized representation for process models in the field of process mining (Boltenhagen et al., 2019).

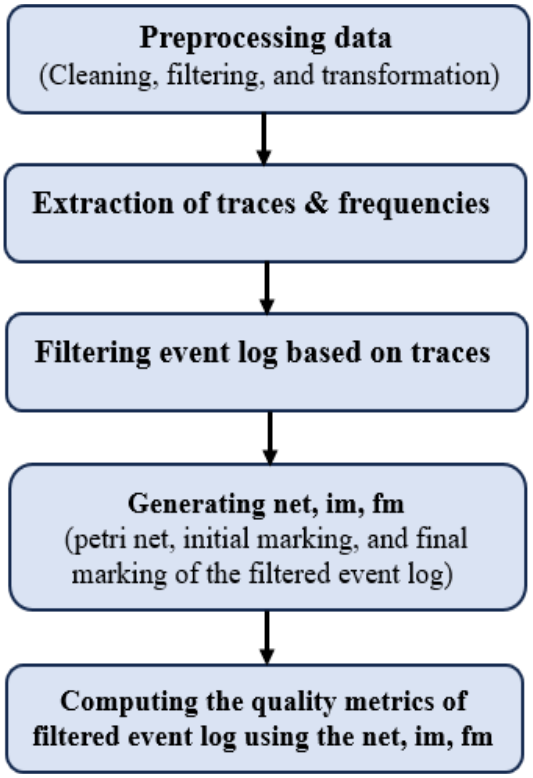

Figure 3. the proposed method

According to Figure 3, in the first step, the event log related to the business process is initially cleared and filtered. Subsequently, the data is transformed into a dataframe in preparation for the next step, which involves extracting the traces and related frequencies of the event log. In the next step, based on the filtered log, a Petri net is constructed along with an initial marking and final marking, which are necessary for computing the quality metrics.

In the fourth step of the proposed method, Petri nets come into play because of their ability to encapsulate the behavior of an accepting Petri net, which includes all traces starting at the initial marking and ending at the final marking. Additionally,

Petri nets are used as the selected method for representing the discovered process models. This choice is attributed to its brevity and straightforward, unambiguous semantics. In the realm of process mining, Petri nets stand as the most widely adopted representation, forming the basis for process models in both Business Process Management (BPM) and Workflow Management (WFM) systems (Van der Aalst, 2016).

4. Findings and Discussion

Considering the event log of the BPI Challenge 2019 (Purchase Order Handling Process), the corresponding process model is shown in Figure 4. It is not easy to decipher this complex process model, and readability is quite challenging. This type of process model can make it difficult to extract information and seriously impact its quality.

Regarding the proposed method in the previous section, after cleaning, filtering, and transforming the event log into a dataframe, high-frequency traces from the event logs are extracted, as shown in Tables 1 and 2. The second column in both Table 1 and Table 2 represents the sequence of activities in each trace, while the third and fourth columns display the absolute frequencies and frequency percentages of these traces. Based on the extracted traces, some of the most frequent traces are selected for filtering the event log. The top-n trace filters retain in the log only those cases that follow one of the n most frequently occurring traces.

Before filtering event logs based on high-frequency traces, the process model of event log BPI 2019 is constructed in the form of a Heuristics Net, which is shown in Figure 4. The reason for using Heuristic Nets is that they are frequently used in scenarios that require quick insights into a process, reducing the need for complex manual modeling. They prove particularly valuable during the initial phases of process analysis when the complete process details remain undisclosed (Berti et al., 2019). Moreover, they can cope with incomplete, noisy, or ambiguous event logs. Despite heuristic nets being frequently utilized in situations where quick insights into a process are needed without extensive manual modeling, the structure of the process model in Figure 4 is still complicated. It does not provide a sufficient amount of clear information to make the process model understandable.

Therefore, in such situations, representing subprocesses from event logs involves filtering the logs based on the most frequently occurring traces identified.

Two different filtered event logs are extracted from each of the event logs: BPI 2017-filtered 1 and BPI 2017-filtered 2 (filtered to include 16.4% and 11.4% of high-frequency traces from BPI 2017, respectively); and BPI 2019-filtered 1 and BPI 2019-filtered 2 (filtered to include 67.3% and 57.76% of high-frequency traces from BPI 2019, respectively). These logs are presented in Table 3, with additional details available in the first and second columns of the table. Finally, the filtered logs are utilized for constructing a Petri net, complete with an initial marking and final marking, which are essential for computing quality metrics. The quality metrics are computed for each of the aforementioned filtered event logs, which are shown in the third column of Table 3. The quality metrics include fitness, precision, generalization, and simplicity.

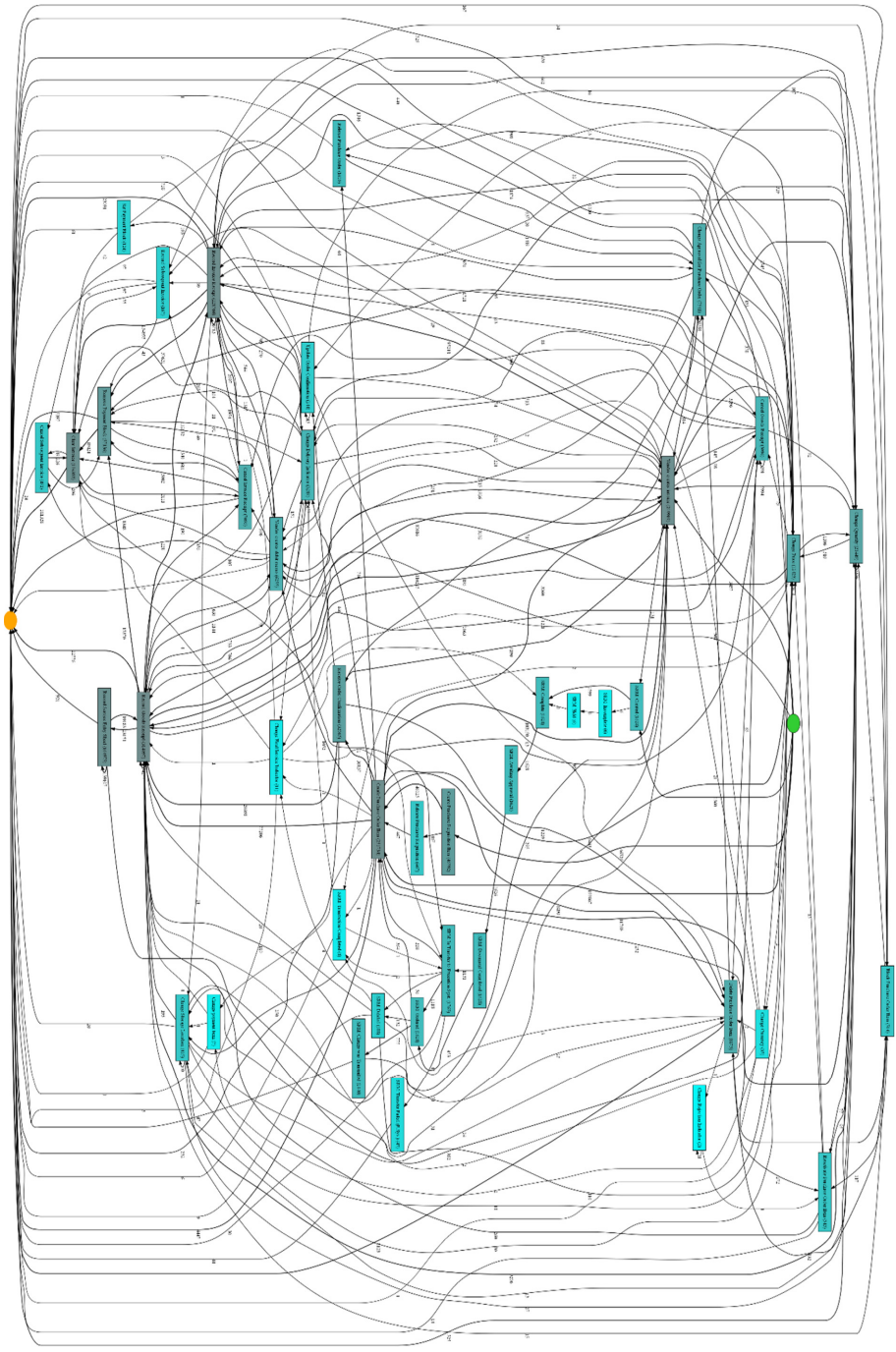

Figure 4. Process model-BPI 2019

Table 1. high frequency traces of BPI 2017

Trace Num.	Trace	Freq	Freq%
1	(‘A_Create Application’, ‘A_Submitted’, ‘W_Handle leads’, ‘W_Handle leads’, ‘W_Complete application’, ‘A_Concept’, ‘W_Complete application’, ‘A_Accepted’, ‘O_Create Offer’, ‘O_Created’, ‘O_Sent (mail and online)’, ‘W_Complete application’, ‘W_Call after offers’, ‘W_Call after offers’, ‘A_Complete’, ‘W_Call after offers’, ‘W_Call after offers’, ‘W_Call after offers’, ‘A_ Cancelled’, ‘O_Cancelled’, ‘W_Call after offers’)	1056	3.351423
2	(‘A_Create Application’, ‘W_Complete application’, ‘W_Complete application’, ‘A_Concept’, ‘A_Accepted’, ‘O_Create Offer’, ‘O_ Created’, ‘O_Sent (mail and online)’, ‘W_Complete application’, ‘W_Call after offers’, ‘W_Call after offers’, ‘A_Complete’, ‘W_Call after offers’, ‘W_Call after offers’, ‘W_Call after offers’, ‘A_ Cancelled’, ‘O_Cancelled’, ‘W_Call after offers’)	1021	3.240344
3	(‘A_Create Application’, ‘A_Submitted’, ‘W_Handle leads’, ‘W_Handle leads’, ‘W_Complete application’, ‘A_Concept’, ‘W_ Complete application’, ‘W_Complete application’, ‘A_Accepted’, ‘O_Create Offer’, ‘O_Created’, ‘O_Sent (mail and online)’, ‘W_ Complete application’, ‘W_Call after offers’, ‘W_Call after offers’, ‘A_Complete’, ‘W_Call after offers’, ‘W_Call after offers’, ‘W_Call after offers’, ‘A_Cancelled’, ‘O_Cancelled’, ‘W_Call after offers’)	734	2.329493
4	(‘A_Create Application’, ‘A_Submitted’, ‘W_Handle leads’, ‘W_Handle leads’, ‘W_Complete application’, ‘A_Concept’, ‘W_ Complete application’, ‘W_Complete application’, ‘W_Complete application’, ‘A_Accepted’, ‘O_Create Offer’, ‘O_Created’, ‘O_Sent (mail and online)’, ‘W_Complete application’, ‘W_Call after offers’, ‘W_Call after offers’, ‘A_Complete’, ‘W_Call after offers’, ‘W_Call after offers’, ‘W_Call after offers’, ‘A_Cancelled’, ‘O_Cancelled’, ‘W_Call after offers’)	451	1.431337
5	(‘A_Create Application’, ‘A_Submitted’, ‘W_Handle leads’, ‘W_Handle leads’, ‘W_Complete application’, ‘A_Concept’, ‘A_Accepted’, ‘O_Create Offer’, ‘O_Created’, ‘O_Sent (mail and online)’, ‘W_Complete application’, ‘W_Call after offers’, ‘W_Call after offers’, ‘A_Complete’, ‘W_Call after offers’, ‘W_Call after offers’, ‘W_Call after offers’, ‘A_Cancelled’, ‘O_Cancelled’, ‘W_Call after offers’)	332	1.053667
6	(‘A_Create Application’, ‘A_Submitted’, ‘W_Handle leads’, ‘W_Handle leads’, ‘W_Complete application’, ‘A_Concept’, ‘W_ Complete application’, ‘W_Complete application’, ‘W_Complete	298	0.945762

Trace Num.	Trace	Freq	Freq%
	application', 'W_Complete application', 'A_Accepted', 'O_Create Offer', 'O_Created', 'O_Sent (mail and online)', 'W_Complete application', 'W_Call after offers', 'W_Call after offers', 'A_ Complete', 'W_Call after offers', 'W_Call after offers', 'W_Call after offers', 'A_Cancelled', 'O_Cancelled', 'W_Call after offers')		
7	('A_Create Application', 'W_Complete application', 'W_Complete application', 'A_Concept', 'A_Accepted', 'O_Create Offer', 'O_ Created', 'O_Sent (mail and online)', 'W_Complete application', 'W_Call after offers', 'W_Call after offers', 'A_Complete', 'W_Call after offers', 'W_Call after offers', 'W_Call after offers', 'W_Call after offers', 'W_Validate application', 'W_Validate application', 'A_ Validating', 'O_Returned', 'W_Validate application', 'O_Accepted', 'A_Pending', 'W_Validate application')	278	0.882288
8	('A_Create Application', 'A_Submitted', 'W_Handle leads', 'W_Handle leads', 'W_Complete application', 'A_Concept', 'W_Complete application', 'A_Accepted', 'O_Create Offer', 'O_Created', 'O_Sent (mail and online)', 'W_Complete application', 'W_Call after offers', 'W_Call after offers', 'A_Complete', 'W_Call after offers', 'W_Call after offers', 'W_Call after offers', 'W_Call after offers', 'W_Validate application', 'W_Validate application', 'A_ Validating', 'O_Returned', 'W_Validate application', 'O_Accepted', 'A_Pending', 'W_Validate application')	244	0.774382
9	('A_Create Application', 'A_Submitted', 'W_Handle leads', 'W_Handle leads', 'W_Complete application', 'A_Concept', 'W_ Complete application', 'W_Complete application', 'A_Accepted', 'O_Create Offer', 'O_Created', 'O_Sent (mail and online)', 'W_Complete application', 'W_Call after offers', 'W_Call after offers', 'A_Complete', 'W_Call after offers', 'W_Call after offers', 'W_Call after offers', 'W_Call after offers', 'W_Validate application', 'W_Validate application', 'A_Validating', 'O_Returned', 'W_Validate application', 'O_Accepted', 'A_Pending', 'W_Validate application')	212	0.672824
10	('A_Create Application', 'A_Submitted', 'W_Handle leads', 'W_Handle leads', 'W_Complete application', 'A_Concept', 'W_Complete application', 'W_Complete application', 'W_Complete application', 'W_ Complete application', 'W_Complete application', 'A_Accepted', 'O_ Create Offer', 'O_Created', 'O_Sent (mail and online)', 'W_Complete application', 'W_Call after offers', 'W_Call after offers', 'A_Complete', 'W_Call after offers', 'W_Call after offers', 'W_Call after offers', 'A_ Cancelled', 'O_Cancelled', 'W_Call after offers')	204	0.647434

Table 2. high frequency traces of BPI 2019

Trace Num.	Trace	Freq	Freq%
1	(‘Create Purchase Order Item’, ‘Vendor creates invoice’, ‘Record Goods Receipt’, ‘Record Invoice Receipt’, ‘Clear Invoice’)	50286	19.97585
2	(‘Create Purchase Order Item’, ‘Record Goods Receipt’, ‘Vendor creates invoice’, ‘Record Invoice Receipt’, ‘Clear Invoice’)	30798	12.23434
3	(‘Create Purchase Order Item’, ‘Record Goods Receipt’)	12214	4.851947
4	(‘Create Purchase Order Item’, ‘Vendor creates invoice’, ‘Record Goods Receipt’, ‘Record Invoice Receipt’, ‘Remove Payment Block’, ‘Clear Invoice’)	11383	4.521837
5	(‘Create Purchase Order Item’, ‘Receive Order Confirmation’, ‘Record Goods Receipt’, ‘Vendor creates invoice’, ‘Record Invoice Receipt’, ‘Clear Invoice’)	9694	3.85089
6	(‘Create Purchase Requisition Item’, ‘Create Purchase Order Item’, ‘Vendor creates invoice’, ‘Record Goods Receipt’, ‘Record Invoice Receipt’, ‘Clear Invoice’)	8921	3.54382
7	(‘Create Purchase Order Item’, ‘Vendor creates invoice’, ‘Record Invoice Receipt’, ‘Record Goods Receipt’, ‘Remove Payment Block’, ‘Clear Invoice’)	8835	3.509657
8	(‘Create Purchase Order Item’, ‘Record Goods Receipt’, ‘Vendor creates invoice’, ‘Record Invoice Receipt’, ‘Remove Payment Block’, ‘Clear Invoice’)	7985	3.171999
9	(‘Create Purchase Order Item’, ‘Delete Purchase Order Item’)	5298	2.104602
10	(‘Create Purchase Order Item’, ‘Receive Order Confirmation’, ‘Vendor creates invoice’, ‘Record Goods Receipt’, ‘Record Invoice Receipt’, ‘Clear Invoice’)	4244	1.685907
11	(‘Create Purchase Requisition Item’, ‘Create Purchase Order Item’, ‘Vendor creates invoice’, ‘Record Goods Receipt’, ‘Record Invoice Receipt’)	4210	1.6724
12	(‘Create Purchase Requisition Item’, ‘Create Purchase Order Item’, ‘Record Goods Receipt’)	3723	1.478942
13	(‘Create Purchase Order Item’, ‘Record Goods Receipt’, ‘Vendor creates invoice’, ‘Record Invoice Receipt’)	3548	1.409424
14	(‘Create Purchase Order Item’,)	2835	1.126189
15	(‘Create Purchase Order Item’, ‘Vendor creates invoice’, ‘Record Goods Receipt’, ‘Record Invoice Receipt’)	2765	1.098382

Trace Num.	Trace	Freq	Freq%
16	(‘Create Purchase Requisition Item’, ‘Create Purchase Order Item’, ‘Record Goods Receipt’, ‘Vendor creates invoice’, ‘Record Invoice Receipt’, ‘Clear Invoice’)	2694	1.070177
17	(‘Create Purchase Requisition Item’, ‘Create Purchase Order Item’, ‘Receive Order Confirmation’, ‘Record Goods Receipt’, ‘Vendor creates invoice’, ‘Record Invoice Receipt’)	2221	0.882281
18	(‘Create Purchase Requisition Item’, ‘Create Purchase Order Item’, ‘Record Goods Receipt’, ‘Vendor creates invoice’, ‘Record Invoice Receipt’)	2054	0.815941
19	(‘Create Purchase Order Item’, ‘Change Quantity’, ‘Vendor creates invoice’, ‘Record Goods Receipt’, ‘Record Invoice Receipt’, ‘Clear Invoice’)	1898	0.75397
20	(‘Create Purchase Requisition Item’, ‘Create Purchase Order Item’, ‘Vendor creates invoice’, ‘Record Goods Receipt’, ‘Record Invoice Receipt’, ‘Remove Payment Block’, ‘Clear Invoice’)	1871	0.743245

Table 3. Quality metrics of the event logs

Event log	Traces for filtering (percentage)	Quality metrics			
		Fitness	Precision	Generalization	Simplicity
BPI 2017	No filtering	0.963	0.813	0.957	0.514
BPI 2017-filtered 1	16.4%	0.984	0.826	0.976	0.634
BPI 2017-filtered 2	11.4%	0.983	0.802	0.983	0.686
BPI 2019	No filtering	0.915	0.872	0.870	0.467
BPI 2019-filtered 1	67.3%	0.958	0.972	0.928	0.609
BPI 2019-filtered 2	57.76%	0.958	0.965	0.994	0.696

As shown in Table 3, in BPI 2017-filtered 2, despite using a lower percentage of traces for log filtering, its precision is lower than that of the original log (Precision value for BPI 2017-filtered 2 is 0.802, whereas it is 0.813 in the original log). In contrast, BPI 2017-filtered 1, which employed a higher percentage of traces for filtering compared to BPI 2017-filtered 2, achieved a better precision value.

Overall, BPI 2017-filtered 2 and BPI 2019-filtered 1 exhibit better quality metric values than their respective original logs within their groups. Consequently,

it can be inferred that the process models derived from BPI 2017-filtered 2 and BPI 2019-filtered 1 possess relatively higher quality metric values when compared to their original logs.

The process model of event log BPI 2019-filtered 1 is construced in the form of Heuristics Net, whis is shown in Figure 5. BPI 2019-filtered 1 exhibits superior quality metric values compared to its original logs. In this case, the resulting process model is significantly more comprehensible than the original model depicted in Figure 4. Moreover, reagrding Figure 5, the layout of the process model offers sufficient, clear information to ensure user understanding when compared to the process model in Figure 4.

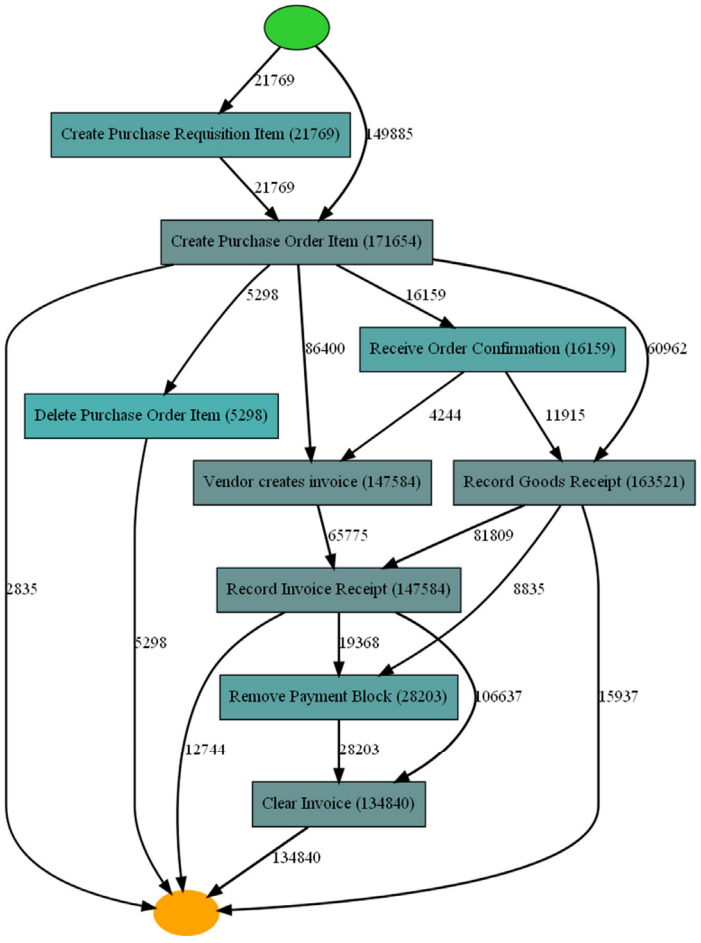

Figure 5. Process model-BPI 2019-filtered 1

High-frequency traces, extracted from the event logs, represent sequences of activities that occur frequently. They play a crucial role in refining complex process models, enhancing readability, and contributing to the overall quality of the model. The top-n trace filtering technique retains only the most frequently occurring traces, providing a focused lens through which the event log is analyzed. Therefore, high-frequency traces and the filtering process are very important for improving process model quality.

Heuristic nets are chosen for constructing process models. Heuristic nets prove valuable in scenarios that require rapid insights into a process, especially during initial phases when complete process details are undisclosed. Their ability to handle incomplete, noisy, or ambiguous event logs is highlighted. Despite the advantages, the complexity of the resulting Heuristic Net in Figure 4 necessitates further refinement through trace filtering.

Two sets of filtered event logs are generated from BPI 2017 and BPI 2019, each representing different proportions of high-frequency traces. The rationale behind these percentages is to balance the inclusion of significant traces while managing the complexity of the logs. These filtered logs serve as the foundation for constructing Petri nets.

The evaluation of quality metrics for each filtered event log is explained, focusing on fitness, precision, generalization, and simplicity. This emphasizes the significance of these metrics in assessing the overall quality and effectiveness of the process models. A comparative analysis between the original logs and their filtered versions, particularly emphasizing cases where lower percentages result in improved precision, provides insights into the subtle effects of filtering strategies.

Despite using lower percentages for log filtering, BPI 2017-filtered 2 exhibits lower precision than the original log, highlighting the complexities of the filtering process. It emphasizes how specific filtering strategies, even with lower percentages, can yield superior precision. This nuanced understanding of the impact of filtering strategies on model quality informs practitioners and researchers in selecting appropriate approaches.

The finding underscores how the BPI 2019-filtered model, despite filtering a higher percentage of traces, results in a more comprehensible process model compared to the original BPI 2019 model in Figure 4. This insight reinforces

the practical significance of trace filtering in enhancing the clarity and user understanding of process models.

5. Conclusion

In conclusion, this paper explores the critical evaluation of quality metrics for business process event logs, with a particular focus on high-frequency traces to improve process model precision and clarity. The study acknowledges the crucial role of event logs in Process Mining, highlighting their importance as a fundamental data source for extracting insights into process performance and identifying opportunities for improvement. The transition from data quality challenges in event logs to the complexity of process models underscores the importance of addressing issues such as missing and imprecise attribute values, timestamps, and the overall intricacies of constructing and comprehending complex process models. Notably, the study demonstrates that even with lower percentages, specific filtering strategies can yield superior precision, emphasizing the importance of thoughtful approaches in enhancing model quality. The comparative analysis between original logs and their filtered counterparts provides valuable insights into the effectiveness of trace filtering in enhancing process model clarity. The case of BPI 2019-filtered 1, exhibiting a more comprehensible process model despite filtering a higher percentage of traces, underscores the practical significance of trace filtering in enhancing user understanding.

In essence, this paper contributes to the ongoing discourse on process mining and modeling by emphasizing the critical role of high-frequency traces in refining complex process models. By evaluating quality metrics and providing practical insights, the study assists practitioners and researchers in adopting effective strategies to enhance the precision, clarity, and overall quality of business process event logs.

Limitations:

The study has several limitations that warrant consideration. First, the complexity of the chosen high-frequency trace filtering strategy may not be universally applicable, and its effectiveness could depend on the specific characteristics of event logs and processes. Additionally, assumptions about the homogeneity of

high-frequency traces might oversimplify the nuanced nature of process patterns. The study acknowledges the complexity of Heuristic Nets in capturing certain process details but does not thoroughly explore scenarios where these nets might fall short.

Future Research Directions:

Future research could delve into optimal trace filtering strategies, comparing various approaches and assessing the impact of filtering parameters on model quality. Exploring domain-specific considerations and extending the analysis to diverse industries would enhance the generalizability of findings. Investigating dynamic filtering techniques that adapt to evolving process behaviors over time could improve the robustness of the approach. Additionally, future studies might incorporate user-centric evaluations, gathering feedback on model comprehension and practical utility. Addressing the challenge of complex process models, especially when using Heuristic Nets, and conducting benchmarking studies against alternative methodologies would offer a more comprehensive understanding of the proposed approach's strengths and weaknesses.

References

AliAbadi, A., & Mohammadi, M. (2018). A Method for Data Integration in Enterprises Using Web Service. *Iranian Journal of Information Processing and Management*, *33* (4), 1637-1658. DOI: 10.35050/JIPM010.2018.028

Andrews, R., Suriadi, S., Ouyang, C., & Poppe, E. (2018). Towards event log querying for data quality: Let's start with detecting log imperfections. In *On the Move to Meaningful Internet Systems. OTM 2018 Conferences: Confederated International Conferences: CoopIS, C&TC, and ODBASE 2018, Valletta, Malta, October 22-26, 2018, Proceedings, Part I* (pp. 116-134). Springer International Publishing. DOI: 10.1007/978-3-030-02610-3_7

Andrews, R., Wynn, M. T., Vallmuur, K., Ter Hofstede, A. H., Bosley, E., Elcock, M., & Rashford, S. (2019). Leveraging data quality to better prepare for process mining: an approach illustrated through analysing road trauma pre-hospital retrieval and transport processes in Queensland. *International journal of environmental research and public health*, *16* (7), 1138. DOI: 10.3390/ijerph16071138

Andrews, R., van Dun, C. G., Wynn, M. T., Kratsch, W., Röglinger, M. K. E., & ter Hofstede, A. H. (2020). Quality-informed semi-automated event log generation for process mining. *Decision Support Systems*, *132*, 113265. DOI: 10.1016/j.dss.2020.113265

Andrews, R., Emamjome, F., ter Hofstede, A. H., & Reijers, H. A. (2022). Root-cause analysis of process-data quality problems. *Journal of Business Analytics*, *5* (1), 51-75. DOI: 10.1080/2573234X.2021.1947751

Berti, A., van Zelst, S., & Schuster, D. (2023). PM4Py: A process mining library for Python. *Software Impacts*, *17*, 100556. DOI: 10.1016/j.simpa.2023.100556

Boltenhagen, M., Chatain, T., & Carmona, J. (2019). Generalized alignment-based trace clustering of process behavior. In *Application and Theory of Petri Nets and Concurrency: 40th International Conference, PETRI NETS 2019, Aachen, Germany, June 23–28, 2019, Proceedings 40* (pp. 237-257). Springer International Publishing. DOI: 10.1007/978-3-030-21571-2_1

Buijs, J. C., van Dongen, B. F., & van der Aalst, W. M. (2014). Quality dimensions in process discovery: The importance of fitness, precision, generalization and simplicity. *International Journal of Cooperative Information Systems*, *23* (01), 1440001. DOI: 10.1142/S0218843014400012

Ceravolo, P., Damiani, E., Torabi, M., & Barbon, S. (2017). Toward a new generation of log pre-processing methods for process mining. In *Business Process Management Forum: BPM Forum 2017, Barcelona, Spain, September 10-15, 2017, Proceedings 15* (pp. 55-70). Springer International Publishing. DOI: 10.1007/978-3-319-65015-9_4

Chapela-Campa, D., Mucientes, M., & Lama, M. (2019). Mining frequent patterns in process models. *Information Sciences*, *472*, 235-257. DOI: 10.1016/j.ins.2018.09.011

Dakic, D., Stefanovic, D., Vuckovic, T., Zizakov, M., & Stevanov, B. (2023). Event Log Data Quality Issues and Solutions. *Mathematics*, *11* (13), 2858. DOI: 10.3390/math11132858

De San Pedro, J., Carmona, J., & Cortadella, J. (2015). Log-based simplification of process models. In *Business Process Management: 13th International Conference, BPM 2015, Innsbruck, Austria, August 31--September 3, 2015, Proceedings 13* (pp. 457-474). Springer International Publishing. DOI: 10.1007/978-3-319-23063-4_30

Fahland, D., & Van Der Aalst, W. M. (2011). Simplifying mined process models: An approach based on unfoldings. In *Business Process Management: 9th International Conference, BPM 2011, Clermont-Ferrand, France, August 30-September 2, 2011. Proceedings 9* (pp. 362-378). Springer Berlin Heidelberg. DOI: 10.1007/978-3-642-23059-2_27

Fischer, D. A., Goel, K., Andrews, R., van Dun, C. G. J., Wynn, M. T., & Röglinger, M. (2020). Enhancing event log quality: Detecting and quantifying timestamp imperfections. In *Business Process Management: 18th International Conference, BPM 2020, Seville, Spain, September 13–18, 2020, Proceedings 18* (pp. 309-326). Springer International Publishing. DOI: 10.1007/978-3-030-58666-9_18

Ferreira, D. R. (2017). *A primer on process mining: Practical skills with python and graphviz*. Cham: Springer International Publishing. DOI: 10.1007/978-3-030-41819-9

Goel, K., Leemans, S. J., Martin, N., & Wynn, M. T. (2022). Quality-informed process mining: A case for standardised data quality annotations. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, *16* (5), 1-47.

DOI: 10.1145/3511707

Hasanzadeh, A., Namdarian, L., & Elahi, S. (2012). A model for service oriented architecture (SOA) governance maturity. *Iranian Journal of Information Processing and Management*, *27* (3), 681-697.

Ireddy, A. T., & Kovalchuk, S. V. (2023). An Experimental Outlook on Quality Metrics for Process Modelling: A Systematic Review and Meta Analysis. *Algorithms*, *16* (6), 295. DOI: 10.3390/a16060295

Krogstie, J., & Krogstie, J. (2016). *Quality of business process models* (pp. 53-102). Springer InternationalPublishing.DOI:10.1007/978-3-642-34549-4_6

Leemans, M., & van der Aalst, W. M. (2015). Discovery of frequent episodes in event logs. In *Data-Driven Process Discovery and Analysis: 4th International Symposium, SIMPDA 2014, Milan, Italy, November 19-21, 2014, Revised Selected Papers 4* (pp. 1-31). Springer International Publishing.

Liu, J., Xu, J., Zhang, R., & Reiff-Marganiec, S. (2021). A repairing missing activities approach with succession relation for event logs. *Knowledge and Information Systems*, *63* (2), 477-495. DOI:10.1007/s10115-020-01524-6

Marin-Castro, H. M., & Tello-Leal, E. (2021). Event log preprocessing for process mining: a review. *Applied Sciences*, *11* (22), 10556. DOI: 10.3390/app112210556

Mohammadi, M. (2017). A Review of influencing factors on the quality of business process models. *Journal of Economic & Management Perspectives*, *11* (3), 1833-1840.

Reijers, H. A., Mendling, J., & Recker, J. (2015). Business process quality management. *Handbook on Business Process Management 1: Introduction, Methods, and Information Systems*, 167-185. DOI:10.1002/9781118785317.weom070213

Salehi, A., Aghdasi, M., Khatibi, T., & Sheikhmohammady, M. (2023). A Conceptual Framework for Preprocessing and Improving Quality of Event Log in Process Mining. *Iranian Journal of Information Processing and Management*, *38* (3), 945-979. DOI: 10.3390/app112210556

Suriadi, S., Andrews, R., ter Hofstede, A. H., & Wynn, M. T. (2017). Event log imperfection patterns for process mining: Towards a systematic approach to cleaning event logs. *Information systems*, *64*, 132-150.

DOI: 10.1016/j.is.2016.07.011

Tax, N., Sidorova, N., Haakma, R., & van der Aalst, W. M. (2016). Mining local process models. *Journal of Innovation in Digital Ecosystems*, *3* (2), 183-196. DOI: 10.1016/j.jides.2016.11.001

Van der Aalst, W., Adriansyah, A., & Van Dongen, B. (2012). Replaying history on process models for conformance checking and performance analysis. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, *2* (2), 182-192. DOI: 10.1002/widm.1045

Van der Aalst, Wil MP. 2016. Process Mining - Data Science in Action, Second Edition. Springer. DOI: 10.1007/978-3-662-49851-4

Van der Aalst, W. M. (2013, May). Mediating between modeled and observed behavior: The quest for the "right" process: keynote. In *IEEE 7th International Conference on Research Challenges in Information Science (RCIS)* (pp. 1-12). IEEE. DOI: 10.1109/RCIS.2013.6577675

Van Der Aalst, W., Adriansyah, A., De Medeiros, A. K. A., Arcieri, F., Baier, T., Blickle, T., ... & Wynn, M. (2012). Process mining manifesto. In *Business Process Management Workshops: BPM 2011 International Workshops, Clermont-Ferrand, France, August 29, 2011, Revised Selected Papers, Part I 9* (pp. 169-194). Springer Berlin Heidelberg. DOI: 10.1007/978-3-642-28108-2_19

Mohsen Mohammadi

Graduated from the National University of Malaysia (UKM) with a PhD in Information Technology (Industrial Computing) in 2014. He is currently an assistant professor at Esfarayen University of Technology, Iran.

Process mining, Business processes analysis and modeling, and Information systems design are his favorite research fields.