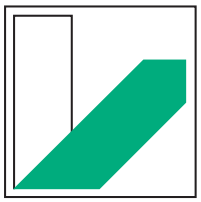


UNIVERSITÄT
BAYREUTH

Artificial Intelligence Along the Business Process Management Lifecycle: Contributions to Process Discovery, Improvement, and Monitoring

Dissertation

zur Erlangung des Grades eines Doktors der Wirtschaftswissenschaft

der Rechts- und Wirtschaftswissenschaftlichen Fakultät

der Universität Bayreuth

vorgelegt

von

Christoph Kecht

aus

Bad Aibling

Dekan:	Prof. Dr. Claas Christian Germelmann
Erstberichterstatter:	Prof. Dr. Maximilian Röglinger
Zweitberichterstatterin:	Prof. Dr. Moe Thandar Wynn
Tag der mündlichen Prüfung:	26.11.2025

ABSTRACT

In a continuously changing world, even well-designed business processes require ongoing improvement to keep pace with evolving customer needs, new technologies, regulations, and market conditions. Organizations must treat Business Process Improvement (BPI) as a continuous effort rather than a one-off initiative to maintain operational excellence and competitive advantage. While process mining can identify inefficiencies from process execution data captured in event logs, turning diagnosed weaknesses into redesign solutions remains largely manual, expertise-intensive, and time-consuming, which leaves a persistent gap between detecting problems and knowing how to fix these problems effectively.

Realizing continuous BPI is further hindered by fragmented improvement knowledge, the prevalence of multiple interrelated weaknesses, and incomplete transparency of the as-is process when key evidence is buried in unstructured data sources such as emails, chats, and documents. Recent advances in Artificial Intelligence (AI) and generative AI, particularly Large Language Models (LLMs), open new opportunities for process analysis and redesign, but must be guided to avoid incorrect or uncontrollable outputs. As organizations embed AI into operational processes, they must also ensure conformance to business rules and robust performance. This thesis addresses these challenges through three research objectives realized in six Research Articles (RAs).

Research Objective 1 focuses on AI-enabled process discovery from unstructured data by automatically constructing event logs from textual data. RA 1 introduces a natural language inference pipeline that extracts topics and process activities from customer service conversations and exports case-centric event logs, allowing large-scale discovery of customer-centric process flows without extensive labeled data. RA 2 extends this approach to object-centric event logs using a two-stage pipeline with a collector and a refiner implemented in heuristic and generative variants. Among four pairwise combinations of collector and refiner instances, the configurations with a generative collector achieve the highest extraction quality, with the fully generative variant in particular producing coherent and standardized event and object labels.

Research Objective 2 provides AI-driven support for process analysis and redesign to reduce reliance on human-centric ideation. RA 3 presents the Process Improvement Copilot, a retrieval-augmented generation-enhanced LLM-based process improvement and innovation system that generates context-specific process improvement ideas grounded in best practices and automatically derived inefficiencies. In expert interviews and a workshop at a multinational technology conglomerate, participants rated the Process Improvement Copilot as useful and easy to use. A substantial share of the generated ideas was considered directly relevant or actionable stimuli for follow-up steps. RA 4 introduces the Automated Business Process Optimizer (ABuPrOpt), which leverages LLMs and simulation to propose and quantitatively evaluate improved to-be process models un-

der consideration of standard and custom improvement objectives. Across five public datasets, ABuPrOpt provided plausible redesign options by generating sound and feasible process models that outperform their original counterparts.

Research Objective 3 advances AI-supported process monitoring for conformance and performance insights. RA 5 develops a chatbot evaluation framework that converts customer service conversations into event logs using the pipeline developed in RA 1 and then applies process mining conformance metrics to quantify how well a trained chatbot adheres to business processes in the underlying training data. RA 6 examines recruitment processes by comparing linguistic analysis of applicants' written self-descriptions with a personality questionnaire in a controlled online experiment with 400 participants. Under salient incentives to fake being cooperative, the text-based AI model significantly outperforms the questionnaire in predicting true cooperativeness and detects signs of exaggeration.

In summary, this thesis contributes approaches integrating AI along the Business Process Management lifecycle. These approaches enable full process transparency by extracting event logs from unstructured data, provide support to generate and evaluate to-be process models with quantitative evidence, and establish monitoring approaches that benchmark AI-enabled operations for conformance and performance. By connecting process discovery, analysis and redesign, and monitoring with end-to-end AI support, the thesis advances the scalability, efficiency, and reliability of continuous BPI.

ACKNOWLEDGEMENTS

This thesis would not have been possible without the guidance, collaboration, and unwavering support of many people to whom I am deeply and sincerely grateful.

First and foremost, I owe my deepest gratitude to my supervisor, Prof. Dr. Maximilian Röglinger. Over the past seven years, your support has reached far beyond research. You helped me grow as a person and as a scholar. Thank you for giving me the freedom to find and follow my own path towards my PhD.

I am profoundly grateful to Prof. Dr. Moe Thandar Wynn from the Queensland University of Technology for your outstanding dedication to our joint research and for serving as the second examiner. Your feedback and encouragement strengthened this work in countless ways.

I sincerely thank Prof. Dr. Niklas Kühl for serving as chairman of the examination committee. I deeply appreciate our collaboration on industry projects and the trust you placed in me throughout these endeavors.

I warmly thank Prof. Dr. Wolfgang Kratsch for your mentorship across our joint research and industry projects and teaching engagements. You supervised me on my first research project when you were a PhD student yourself and sparked a lasting enthusiasm in me for the intersection of computer science and process intelligence.

None of the research articles in this thesis would have been possible without the exceptional collaboration with my co-authors. In alphabetical order, I would like to thank Camille Bitenc, Alina Buss, Franziska Dechert, Dr. Andreas Egger, Prof. Dr. Wolfgang Kratsch, Prof. Dr. Michael Kurschilgen, Prof. Dr. Maximilian Röglinger, Dr. Sareh Sadeghianasl, Uladzimir Smalei, Dr. Magnus Strobel, and Prof. Dr. Moe Thandar Wynn. Working with you was a huge privilege. Your ideas, time, and dedication made these papers possible.

I would also like to extend my heartfelt thanks to my other co-authors with whom I had the pleasure of writing additional research papers: Dr. Felix Baumgarte, Luca Dombetzki, Prof. Dr. Robert Keller, Dr. Daniel Rau, and Linda Wolf. Although our papers are not part of this thesis, our collaborations were immensely valuable in making me a better researcher.

Moreover, I am grateful to my colleagues at the FIM Research Center for Information Management (supported by the University of Bayreuth and the Technical University of Applied Sciences Augsburg) and the Fraunhofer Institute for Applied Information Technology FIT. Listing everyone would be impossible, but working with you on applied research projects was a great joy. I also want to thank our industry partners, notably the participants from the research project "Next Best Process" (funded by the Bavarian Research Foundation), for trusting us with their challenges and for their invaluable feedback on our research.

My sincere thanks go to my mentors at Roland Berger for their continuous support and for enabling me to join the PhD program. In particular, I would like to thank Juliane Zahel, Klaus Fuest, and Dr. Christian Krys. I am equally grateful to my colleagues in the PhD program for the valuable discussions we shared in our seminars.

My deepest gratitude belongs to my family and friends. To my parents and grandparents: thank you for setting aside so much, making sacrifices so my brothers and I could pursue our degrees, and encouraging us to pursue an excellent education. To my friends: thank you for believing that this thesis might advance the world, or for recognizing earlier than I did that its impact may be modest; either way, your good humor and companionship kept me going. To my brothers: thank you for all your support, and for making it possible for me to meet Anika.

Finally, Anika: thank you for showing me that there is so much more to life than writing research papers, and for believing in me when I doubted myself. Our moments together give me the energy I need, and your delicious cooking (in generous portions!) certainly helps. I admire your dedication to your continuing professional development. Thank you for being in my life. I am sure we still have so many wonderful things ahead of us.

Bayreuth, March 2026

Christoph Kecht

“We’ve always done it this way.”

Table of Contents

Copyright Statement

The following sections are partly comprised of content taken from the research articles included in this thesis. To improve the readability of the text, I have omitted the standard labeling of these citations.

1 Introduction

1.1 Motivation

> *"Every good process eventually becomes a bad process."*
>
> – Hammer and Champy (1993, p. 12)

In a continuously changing world, even the most effective business process cannot remain in use forever. As customer needs evolve, technologies advance, regulations shift, and market conditions fluctuate, organizations must not treat process improvement as a one-off endeavor but rather as a continuous, long-term effort (Davenport, 1993; Hammer and Champy, 1993; Huang et al., 2015). Business Process Improvement (BPI) has thus become an essential capability in dynamic environments, ensuring that processes are regularly adapted and optimized to align with new conditions and strategic goals (Blocker et al., 2011; Hosseini et al., 2018; Kerpedzhiev et al., 2021; Kreuzer et al., 2020). Noting that organizations which fail to continuously improve their processes risk performance declines and competitive disadvantage, extant research on Business Process Management (BPM) stresses that maintaining process excellence requires ongoing care and proactive change, echoing the need for a culture of continuous improvement (Dumas et al., 2018; Grisold et al., 2021; Kettinger et al., 1997; Rosemann and vom Brocke, 2015; Zellner, 2011).

However, achieving continuous BPI is challenging (Park and van der Aalst, 2022). While process mining has simplified the detection of process inefficiencies, the subsequent steps, i. e., formulating improvement ideas and designing better process models, remain largely manual, expertise-intensive, and time-consuming endeavors (Groß et al., 2024; Gross et al., 2019; Huang et al., 2015; Limam Mansar et al., 2009). Process inefficiencies occur if a business process meets its operational goals but is wasteful in terms of utilized resources (Wastell et al., 1994). Turning a diagnosed inefficiency into a well-founded redesign solution has traditionally relied on the creativity and experience of domain experts (Figl and Recker, 2016; Mustansir et al., 2022). Therefore, BPI projects often convene process analysts and domain experts in workshops to develop solutions, for example, via techniques like brainstorming (Kettinger et al., 1997) or consulting best practices, because few tasks in the BPM lifecycle are as creativity-intensive as process redesign (Gross et al., 2019). The reliance on human expertise makes BPI initiatives resource-demanding, as they consume scarce expert time, require deep domain and methodological knowledge, and typically must iteratively evaluate multiple redesign alternatives to find a truly effective solution (Beerepoot et al., 2019; Limam Mansar et al., 2009). Indeed, empirical studies consistently report that BPI efforts are costly in terms of personnel and effort, often stretching over long periods to rigorously assess the impact of changes (Bader et al., 2023; Rich and Bateman, 2003).

A further challenge lies in the fragmentation of BPI knowledge. Although a rich body of process improvement knowledge has accumulated in the form of BPI patterns, best practices, and documented case studies (Falk et al., 2013; Kettinger et al., 1997; Malinova et al., 2022; Reijers and Liman Mansar, 2005; Zellner, 2011), this knowledge is often underutilized in existing Process Im-

provement and Innovation Systems (PIISs), resulting in overlooking proven solutions (Vanwersch et al., 2016). The matching between an identified inefficiency and a suitable improvement idea is also nontrivial (Fehrer et al., 2022; Lashkevich et al., 2023; Niedermann and Schwarz, 2011; Park and van der Aalst, 2022; Truong and Lê, 2016). Without computational support, analysts must rely on intuition to connect a process inefficiency with a specific redesign solution. This connection might not be obvious, especially when the best solution comes from a different domain or requires thinking outside conventional patterns. The situation is even more complicated when multiple process weaknesses are present at once (Li et al., 2023; Tang et al., 2023). In real-world BPI initiatives, it is common to discover numerous pain points that are interrelated, such as delays in one part of the process and quality issues in another (Lehnert et al., 2016; Li et al., 2023; Tang et al., 2023). Addressing these in isolation can lead to fragmented fixes that suboptimize one part of the process at the expense of another (Jansen-Vullers et al., 2008a,b; Reijers and Liman Mansar, 2005). Instead, a coherent set of improvement ideas is needed to tackle multiple inefficiencies holistically. Generating such a coordinated improvement plan puts even more cognitive demand on experts, who must consider complex dependencies and avoid solutions that conflict with each other. In summary, despite decades of BPM research and practice, there remains a significant gap between knowing that a process has problems and knowing how to fix those problems effectively (Vanwersch et al., 2016).

Crucially, process transparency is indispensable for BPI. Any improvement initiative must start from a clear understanding of the current process and its weaknesses (Vanwersch et al., 2016; Wastell et al., 1994). Analysts need objective evidence of how work is actually being done, where delays or errors occur, and how far the real execution strays from the intended design (König et al., 2019). In the past, such transparency was challenging to achieve, as process knowledge often resided in the heads of employees or in scattered documents, leading to an incomplete or biased view of what was happening on the ground (Geeganage et al., 2022; Kratsch et al., 2022). Today, the widespread availability of event logs and the maturation of process mining have transformed this situation (Andrews et al., 2020; Calvanese et al., 2016; Schönig et al., 2016). Event logs record the sequence of activities executed in process instances, enabling data-driven discovery of the as-is process model and pinpointing of inefficiencies (van der Aalst, 2012, 2016; van der Aalst et al., 2012; van der Aalst, 2011). Studies have shown that an accurate as-is model is critical for BPI since redesign efforts that skip proper analysis risk addressing symptoms rather than root causes or even introducing new bottlenecks (Al-Mashari and Zairi, 1999; Vanwersch et al., 2016). Therefore, ensuring transparency into the as-is process is a fundamental prerequisite for effective improvement.

Just as data availability and transparency have improved the diagnosis of process inefficiencies, recent advances in Artificial Intelligence (AI) have created new opportunities to support other phases of BPI, especially the generation of improvement ideas, redesign of process models, and even the automation of operational work itself (Feuerriegel et al., 2024). In particular, the rise of Natural Language Processing (NLP) and Generative AI (GenAI) based on Large Language Models (LLMs)

offers powerful new tools for BPI initiatives. LLMs are AI models trained on vast corpora of text, enabling them to understand and generate human-like language by combining computational creativity with strong reasoning capabilities (Brown et al., 2020; Kojima et al., 2022; Vaswani et al., 2017; Wolf et al., 2020; Zhao et al., 2025). These features make LLMs a promising technology to assist in creative, knowledge-intensive tasks like business process redesign. In fact, researchers have already begun applying LLMs in various BPM activities, such as explaining process models (Fahland et al., 2025), supporting the creation of process models (Ziche and Apruzzese, 2024), or even modeling processes by themselves from textual descriptions (Kourani et al., 2024; Köpke and Safan, 2025). Unlike a human expert who may only recall a limited set of past projects or best practices, an LLM can be augmented with a vast repository of BPI knowledge and examples. Through approaches like Retrieval-Augmented Generation (RAG), an LLM-based system can retrieve relevant BPI knowledge to generate tailored suggestions for the process at hand (Balaguer et al., 2024; Lewis et al., 2020b; Shuster et al., 2021).

AI-based BPI support is not limited to suggesting improvement ideas but also extends to the design and evaluation of new process models (Fehrer et al., 2022; Mustansir et al., 2022). Another opportunity created by GenAI is the automated generation of improved process models implementing the suggested changes, and the subsequent simulation or quantitative analysis of those redesigned models. Traditionally, once a team comes up with an improvement idea, designing the new process flow and verifying its performance impact can be labor-intensive tasks requiring modeling expertise and tools for process simulation (Grisold et al., 2021; Jansen-Vullers et al., 2008a,b). Recent research demonstrates that GenAI can be leveraged to propose redesigned process models that satisfy certain improvement objectives (Beheshti et al., 2023; van Dun et al., 2023), such as reducing cycle time or cost, while maintaining compliance with business rules and soundness constraints (van Dongen et al., 2006). These AI-generated process models can be evaluated using process simulation techniques to provide objective feedback on their performance (Afflerbach et al., 2017; Fehrer et al., 2022).

From extracting event logs from raw text to suggesting and validating improvements, AI promises to accelerate and enrich every step of the BPI lifecycle. Moreover, AI can be deployed within business operations to automate routine tasks, for example, handling customer inquiries via chatbots (Kecht et al., 2023), or screening documents and communications, thereby not only improving processes from the outside but also actively changing how processes are executed on the inside. However, along with these opportunities come new challenges. When conversational AI agents become part of operational business processes, either as decision support or as autonomous actors, they introduce novel risks and considerations that organizations must address. One major concern is ensuring that AI agents adhere to the predefined business rules and process models rather than taking unintended actions (Kecht et al., 2023). Unlike traditional software systems, which execute predefined workflows, an LLM-driven chatbot in customer service has a high degree of autonomy, thus raising the question of how to guarantee that such agents follow the desired process consistently. Another challenge is maintaining performance and fairness when AI is involved in processes

that affect people, such as recruitment or employee evaluation processes (Kecht et al., 2022). While AI can bring objectivity and consistency to such tasks, it can also inadvertently introduce biases or be manipulated by technically skilled users (Birkeland et al., 2006; Boyd and Pennebaker, 2017; Hancock et al., 2007; Newman et al., 2003; Rosse et al., 1998; Tett and Simonet, 2021).

In conclusion, the need for continuous BPI in dynamic environments is clear and urgent. Organizations must continuously adapt or face decline, as business processes inevitably age and misalign with their environment. While effective in individual cases, the traditional, human-centric approach to BPI struggles with challenges of scale, speed, and knowledge utilization. Therefore, AI-enabled solutions that can leverage data and computational creativity are indispensable in the long run to support or automate crucial BPI tasks. By enhancing process transparency through unstructured textual data, employing GenAI to generate and evaluate improvement ideas, and integrating AI into operational processes, there is an opportunity to significantly accelerate process improvement and innovation. While this thesis is driven by a vision of end-to-end AI-supported BPI, embracing AI in BPM comes with the responsibility to address the resulting new challenges. Ensuring that AI-suggested improvements are trustworthy, that participants in AI-supported processes behave correctly and ethically, and that human experts remain in control of the overall process change trajectory is paramount.

1.2 Research Objectives

Continuous BPI remains vital for organizations to adapt to changing environments (Dumas et al., 2018; Gross et al., 2021; Hammer and Champy, 1993). A transparent understanding of the as-is process is widely recognized as the indispensable first step of any BPI initiative (Dumas et al., 2018; Vanwersch et al., 2016). Only a faithful baseline model of the current process allows diagnosing root causes of inefficiencies and predicting the impact of changes, whereas skipping thorough as-is analysis often leads to wasted effort or new bottlenecks in later improvements (Al-Mashari and Zairi, 1999; Reijers and Liman Mansar, 2005; Rummler and Brache, 2012). However, achieving such transparency is challenging because a substantial share of process-relevant data resides outside of structured enterprise systems in unstructured formats like emails, chats, and documents (Banziger et al., 2018; Jlailaty et al., 2017; van der Aalst and Nikolov, 2008). These sources frequently capture exception handling and workarounds – the situations where processes deviate from their intended path and where improvement opportunities arise (Alter, 2014; König et al., 2019; Rinderle and Reichert, 2006). Traditional process mining techniques cannot tap these unstructured inputs, leaving critical blind spots in understanding the as-is process (Buss et al., 2025; Kecht et al., 2021; Kratsch et al., 2022; König et al., 2025). This gap motivates the need for new methods to discover and model processes from unstructured data at scale.

Therefore, the first research objective of this thesis is to *enable process discovery from unstructured data* by developing methods that automatically construct event logs from textual sources. This objective addresses the lack of insight into process executions hidden in unstructured data. To

this end, we design NLP-based and LLM-based pipelines to transform textual records (e. g., customer service conversations) into event logs, allowing us to uncover a previously untapped level of granularity, thus laying the foundation for as-is process transparency.

Once process inefficiencies are identified, the next challenge is deriving effective improvements and redesigns. Turning diagnosed issues into well-founded improvement ideas and to-be process models remains a heavily manual, creativity-intensive, and resource-demanding endeavor in practice (Figl and Recker, 2016; Groß et al., 2024; Gross et al., 2019; Huang et al., 2015; Mustansir et al., 2022; Reijers and Liman Mansar, 2005). BPI projects consume scarce expert time and require deep domain knowledge, as generating and rigorously evaluating redesign alternatives hinges on human expertise (Bader et al., 2023; Beerepoot et al., 2019; Zellner, 2011). Recent calls for PAISs (Fehrer et al., 2025; Moder et al., 2025) urge the development of (semi-)automated, scalable support for this phase (Beerepoot et al., 2023; Park and van der Aalst, 2022). Yet, current computational approaches typically automate only isolated tasks (e. g., recommending a known redesign pattern or applying a single optimization heuristic) and thus fall short of providing holistic support. These approaches particularly struggle to reuse the rich body of existing BPI knowledge or explore beyond predefined solution options, and they often cannot ensure that redesigned processes are sound (van Dongen et al., 2006) and perform better on key metrics. In summary, there is a clear opportunity for advanced AI techniques to assist process analysts in generating and evaluating BPI ideas. Emerging GenAI, particularly LLMs, offers computational creativity and reasoning capabilities (Kojima et al., 2022; Zhao et al., 2025), but their direct application in BPI is constrained by issues like hallucinations and limited output controllability (Agrawal et al., 2024; Huang et al., 2025; Maynez et al., 2020; Ye et al., 2024). These observations necessitate research into guided AI approaches that can reliably augment human creativity in process analysis and redesign.

Hence, the second research objective of this thesis is to *provide AI-driven support for process analysis and redesign*. This objective targets the analysis and redesign phases of the BPM lifecycle, striving to reduce reliance on human-centric ideation. We develop two complementary artifacts. The first artifact, a RAG-enhanced LLM-based Process Improvement Copilot, draws on event logs and a knowledge base of best practices to automatically suggest context-specific and justified improvement ideas. The second artifact, the Automated Business Process Optimizer (ABuPrOpt) leverages LLMs alongside process simulation to generate and quantitatively evaluate improved process models. Both solutions broaden the space of possible changes beyond conventional heuristics, ensure that any proposed to-be process model is sound and feasible, and evaluate the changes quantitatively, thereby helping to overcome current bottlenecks in BPI.

Furthermore, as organizations increasingly automate and augment processes with AI (Adamopoulou and Moussiades, 2020; Brandtzaeg and Følstad, 2017), new concerns arise regarding compliance and effectiveness. For example, a customer service chatbot might converse fluently, but that alone does not guarantee business value (Kecht et al., 2023). Chatbots must also follow prescribed business process rules, such as authenticating customers before disclosing information (Gunson et al., 2011) to be truly useful. However, most chatbot evaluations today focus almost exclusively on con-

versational quality, lacking systematic measures of whether chatbots' behavior conforms to an organization's business processes. Similarly, in recruitment processes, traditional psychometric tests (e. g., personality questionnaires) are not robust against manipulation when candidates have incentives to fake their answers (Morgeson et al., 2007; Tett and Simonet, 2021). This raises the question of whether AI-based analysis of textual data, for example, applicants' cover letters, can yield more reliable insights into traits like cooperativeness, even under such faking conditions (Boyd and Pennebaker, 2017; Stachl et al., 2020). Since organizations risk deploying AI solutions that violate business rules or produce unreliable performance indications, improved monitoring approaches are required to ensure that AI-driven processes remain compliant and effective.

Consequently, the third objective of this thesis is to advance *AI-supported process monitoring for conformance and performance insights*. This objective focuses on the monitoring phase of the BPM lifecycle, where we deal with evaluating AI systems embedded in business processes. We introduce a novel chatbot evaluation framework that quantifies a chatbot's adherence to business processes by applying process mining conformance metrics. We also investigate an NLP-based approach to personality assessment by analyzing applicants' cover letters to predict cooperativeness, a key behavioral trait, under conditions where subjects are incentivized to fake their true cooperativeness. Together, these studies provide tools to monitor AI-driven processes by yielding objective conformance measures to benchmark and govern chatbot behavior before deployment, and by examining the reliability of AI-driven personality assessment in recruiting.

1.3 Structure of the Thesis and Embedding of the Research Articles

As illustrated in Figure 1, this thesis is structured along the BPM lifecycle of Dumas et al. (2018). The thesis embeds six Research Articles (RAs) that contribute to the phases of process discovery, process analysis and redesign, and process monitoring.

Section 2 focuses on AI-enabled *process discovery* from unstructured data. In this section, RA 1 introduces a Natural Language Inference (NLI)-based pipeline that derives topics and process activities from textual customer service conversations and exports them as case-centric event logs, thus enabling the large-scale discovery of customer-centric process flows that were previously hidden in unstructured communication data. RA 2 extends this contribution by targeting object-centric process data with an approach to extract Object-Centric Event Logs (OCELs) from textual descriptions using a two-stage pipeline which consists of a collector and a refiner, each realized with heuristic and generative variants. Among the four studied configurations, the configurations with a generative collector achieve the highest extraction quality, with the fully generative variant in particular producing coherent and standardized event and object labels, thereby effectively eliminating blind spots and providing richer context for subsequent analysis. Together, RA 1 and RA 2 lay the foundation for process transparency by supplying a robust *as-is process model* for the subsequent phases of the BPM lifecycle and uncovering the situations where processes deviate from their intended sequence and where improvement opportunities arise.

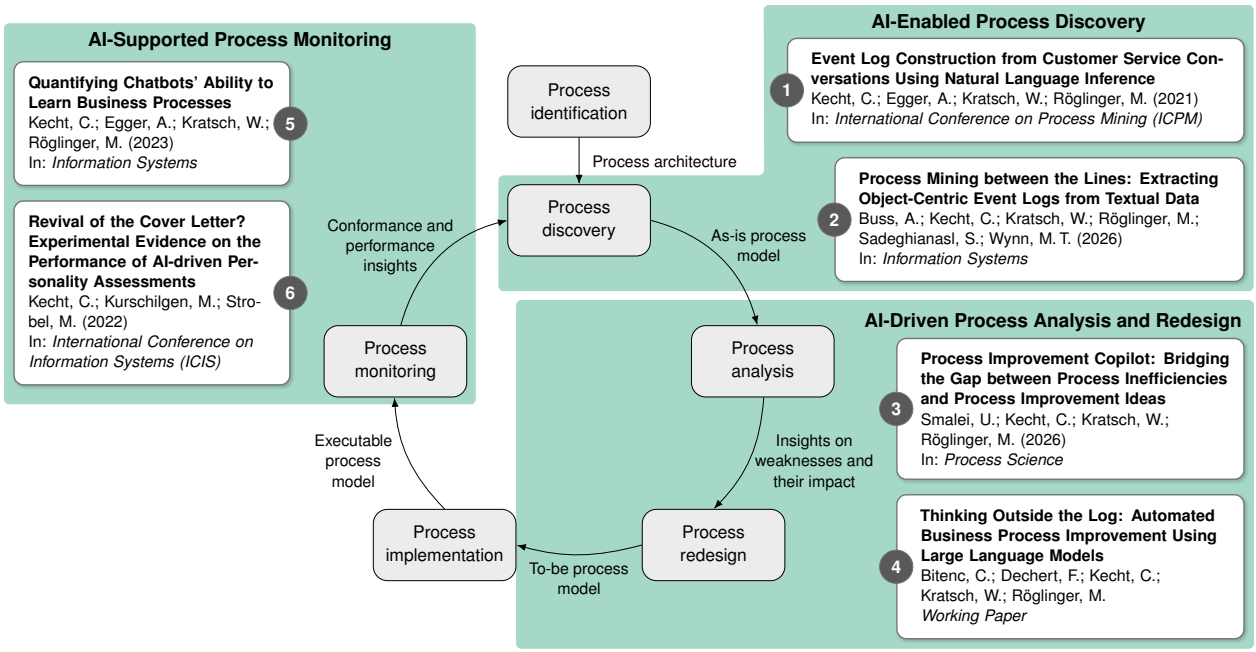


Figure 1: Embedding of the six research articles in this thesis

Section 3 covers AI-driven *process analysis and redesign*. RA 3 presents the Process Improvement Copilot, a RAG-enhanced LLM-based PIIS. This artifact bridges the gap between identified process inefficiencies and actionable improvement ideas as it draws on event logs and a knowledge base of BPI best practices (e. g., patterns and case studies) to automatically generate context-specific improvement suggestions, each accompanied by justifications grounded in the retrieved knowledge. By reducing reliance on scarce expert resources, RA 3 demonstrates how LLMs can serve as a copilot for ideation in BPI initiatives. RA 4 complements this contribution by introducing the ABuPrOpt, which leverages LLMs alongside process simulation techniques to generate and quantitatively evaluate improved process models. ABuPrOpt expands the solution space of possible process redesigns beyond conventional heuristics, incorporates both standard and custom improvement objectives, and simulates the performance of each generated process model to ensure it is not only syntactically correct but also yields better cycle time and execution cost than the existing process model. Through RA 3 and RA 4, Section 3 shows how GenAI can uncover process weaknesses and propose redesigns at scale, thereby producing actionable *insights on weaknesses and their impact* and delivering a high-quality *to-be process model* ready for implementation.

Section 4 addresses AI-supported *process monitoring* in environments where operational processes are increasingly automated or assisted by AI. In RA 5, we develop a novel evaluation framework to quantify chatbots' adherence to business processes. This study develops a chatbot evaluation workflow that first uses the NLI-based extraction pipeline from RA 1 to convert large volumes of customer service conversations into event logs, and then applies process mining conformance checking techniques to measure how well a trained chatbot follows business processes in the underlying training data. The obtained conformance metrics enable an objective benchmarking of

different chatbot versions on their process compliance before deployment, ensuring that conversational AI solutions align with organizational and regulatory process requirements. Finally, RA 6 explores AI-supported process monitoring in recruitment, tackling the challenge of fair and effective personality assessment. This study investigates whether analyzing applicants' free-text self-descriptions (simulated cover letters) with an NLP-based classifier can predict cooperativeness, an important behavioral trait, more reliably than traditional psychometric questionnaires when candidates have incentives to fake their responses. In a controlled online experiment involving 400 participants, the text-based AI model significantly outperforms a standard personality test in identifying truly cooperative behavior under faking conditions. Taken together, RA 5 and RA 6 show how AI-supported process monitoring delivers actionable *conformance and performance insights*, i. e., conformance metrics that benchmark and govern chatbot deployments, and performance evidence that NLP-based personality assessments more robustly identify cooperative applicants than self-reported tests when faking incentives are present.

Section 5 summarizes how AI and GenAI enable end-to-end BPI from event log construction through redesign and simulation to monitoring, and outlines limitations and promising avenues for future work. Appendix A provides an index of the RAs and my individual contribution to each article, and includes bibliographic details along with the corresponding abstracts (or extended abstracts).

2 AI-Enabled Process Discovery

Transparent insight into the as-is state of a business process is widely recognized as the indispensable starting point of any BPI initiative. BPM textbooks and empirical studies alike stress that only a faithful baseline model allows diagnosing root causes of inefficiencies and forecasting the impact of proposed changes (Dumas et al., 2018; Hammer and Champy, 1993; Reijers and Liman Mansar, 2005; Rummler and Brache, 2012). Conversely, redesign projects that skip or shortcut the as-is analysis frequently waste resources on changes with low leverage or introduce new bottlenecks (Al-Mashari and Zairi, 1999; Dumas et al., 2018). Developing such transparency is therefore not optional but a prerequisite for the later analysis, redesign, and monitoring phases in the BPM lifecycle.

Process mining addresses this need by discovering, monitoring, and enhancing business processes based on event logs that capture the sequence and context of activities taken within a business process (van der Aalst, 2012, 2016, 2011). Event logs are traditionally retrieved from structured data held in core information systems using database-oriented extraction techniques (Andrews et al., 2020; Calvanese et al., 2016; Schönig et al., 2016). However, a substantial share of process-relevant data is generated outside those systems in unstructured formats, such as phone calls, e-mails, and contracts (Banziger et al., 2018; Jlailaty et al., 2017; van der Aalst and Nikolov, 2008). These sources often document exception handling and manual workarounds, i. e., exactly the situations where processes deviate from their intended sequence and where improvement opportunities arise (Alter, 2014; König et al., 2019; Rinderle and Reichert, 2006).

Unlocking this untapped information requires NLP and GenAI techniques that transform unstructured text into high-quality event logs. Section 2.1 (RA 1) introduces an NLI pipeline that derives process activities and topics from customer service conversations and exports them as case-centric eXtensible Event Stream (XES) event logs, enabling the large-scale discovery of customer-centric processes. Section 2.2 (RA 2) extends this contribution by focusing on an object-centric representation of textual data: it introduces an OCEL extractor consisting of a collector and a refiner, each instantiated in heuristic and generative variants. Among the resulting four combinations, the configurations with a generative collector achieve the highest extraction quality, with the fully generative variant in particular producing coherent and standardized event and object labels, thereby eliminating blind spots and providing a richer context for subsequent analysis (Geeganage et al., 2022; Grisold et al., 2021; Kratsch et al., 2022). Together, these studies lay the foundations for the AI-enabled process discovery phase of the BPM lifecycle.

2.1 Case-Centric Event Log Construction from Textual Data

To construct event logs from textual data, related approaches already demonstrate the successful application of algorithms in the fields of NLP and machine learning (Banziger et al., 2018; Jlailaty et al., 2017). Machine learning approaches typically require excessive training to achieve high data quality, and thus, a vast amount of labeled training data. However, recent advancements in the

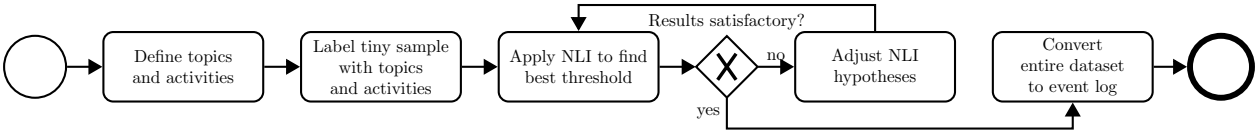


Figure 2: Pipeline for constructing case-centric event logs from customer service conversations

field of NLP research mitigated this issue, among others, by providing pre-trained language models for a multitude of NLP tasks (Lewis et al., 2020a), such as NLI. Given two sentences, referred to as the hypothesis and the premise, NLI determines whether the hypothesis can be inferred from the premise (MacCartney and Manning, 2008). For example, the hypothesis "This sentence is an apology" can be inferred from the premise "We are sorry for the unpleasant experience". Combined with a pre-trained language model, Yin et al. (2019) show the applicability of NLI for topic, emotion, and situation detection.

In RA 1 (Kecht et al., 2021), we utilize NLI to derive topics and process activities from customer service conversations that follow a question-answer pattern and represent them in a case-centric event log, assuming the presence of essential event log attributes (case ID, event ID, and timestamp) in the respective dataset. To this end, we compute the probability that a sentence describing the topic or the process activity can be inferred from the customer's inquiry or the agent's response using NLI. By embedding this concept into a reusable workflow, we develop an approach to represent customer service conversations as standardized IEEE XES event logs (IEEE, 2016), which can then be imported and used by process mining applications, such as ProM (van Dongen et al., 2005) and Disco (Günther and Rozinat, 2012).

Figure 2 provides an overview of our approach for constructing event logs from customer service conversations. Our approach automatically derives topics and activities using NLI and converts other essential event log attributes (case ID, event ID, and timestamp). Since customer service conversations follow a typical question-answer pattern, we map the customer's inquiry to one or many topics and the service agent's answer to one or more activities, which we both represent in the resulting event log in the standardized `case:concept:name` attribute. For a later distinction between the customers' inquiries and the agents' responses, the approach accordingly populates the standardized `org:resource` attribute in the event log.

As Figure 2 outlines, the utilization of the approach comprises four mandatory and one optional step. First, the domain-specific topics categorizing the customer's inquiry as well as the activities describing the service agent's response have to be defined. The topics can include, for example, issues related to the delivery of an order, a particular type of product, or the customer's account. In contrast, examples for the activities include requesting the customer number, apologizing for inconveniences, or asking for specific details.

In the second step, a tiny sample of the customers' inquiries and the service agents' responses are manually labeled using a binary encoding with the true topics and activities, respectively. Since an inquiry can comprise multiple topics and a response can comprise multiple activities, a message

can be assigned to more than one category. A sample size of 100 pairs of customers' inquiries and agents' responses is sufficient to achieve a suitable accuracy when automatically assigning the remaining conversations later on due to the high accuracy of pre-trained language models (as our results indicate).

Afterward, the NLI algorithm is applied in the third step. In our approach, the customer's inquiry or the agent's response is the premise, whereas the hypothesis is a sentence describing the topic of the inquiry or the activity in the response. In this step, we use the default hypothesis "This example is [topic/activity]" of the Python library "transformers" (Wolf et al., 2020), which internally calls a BART model (Lewis et al., 2020a) and yields a probability for each combination of premise and hypothesis. However, since the assignment of inquiries and responses to topics and process activities is a binary decision, we need to determine a robust decision threshold to either assign or not assign the inquiries and responses. To this end, we apply a cross-validation procedure, implemented using the Python machine learning library "scikit-learn" (Pedregosa et al., 2011). For each combination of premise and hypothesis, the respective manually assigned labels from the second step and the probability computed previously are split into five disjoint lists. In each fold, four of the lists serve as the training set, whereas the remaining list serves as the test set and, thus, is used for independent validation. The folds are stratified, i. e., the original list's distribution is maintained across the individual lists. If a label is present less than five times, the number of folds can be reduced accordingly, or the label can be considered irrelevant, and thus, can be dropped. To determine the optimal decision threshold, for each candidate in the set $\{0.70, \ldots, 0.97, 0.980, \ldots, 0.999\}$, we assign all items in each fold's test set to the topic or action if the item's probability is greater or equal than the candidate. The optimal decision threshold for each topic or activity is the candidate that achieves the highest Matthews Correlation Coefficient (MCC) across all folds.

The optional fourth step involves defining further NLI hypotheses that describe the defined topics and activities if the MCCs obtained in the previous step are not satisfactory. This step is optional since, in some cases, the default hypothesis achieves reliable results. However, other candidates, such as "The sentence is about [topic/activity]", "The customer asks about [topic]", and "Please provide/send ..." can lead to significant improvements for some topics and activities. The decision which results are satisfactory depends on the specific use case and remains to the user of our approach.

The remaining fifth step is constructing an event log that contains a trace with a corresponding case identifier for each conversation. Based on the hypotheses and thresholds computed in the previous steps, NLI is applied to all messages for each conversation. If the computed probability is greater or equal to the threshold for the respective topic or activity, the message is classified accordingly. For each assigned topic or activity, an event is inserted into the trace, including the id of the message, the timestamp, the author (in the `org:resource` attribute), the text, and the assigned topic or activity (in the `case:concept:name` attribute). To this end, the Python library "PM4Py" (Berti et al., 2023) provides a function to export a classified dataset of events into a case-centric XES event log.

To evaluate our approach, we compare the performance of the best NLI hypothesis (referred to as "NLI - final" in the following) and the default NLI hypothesis ("NLI - default") for each topic or process activity to the performance of a simple keyword-based classifier ("Keyword"). The latter looks up if a message (converted to lowercase) contains the given keyword. In this case, it assigns the message to the respective activity or topic. For example, if a customer's inquiry contains the word "deliver", the message is assigned to the topic "delivery". Similarly, if a customer service agent's response contains "?", the message is assigned to the activity "Investigate issue". We claim the keyword-based classifier as a suitable benchmark since it is comparatively simple to implement and strains less computational complexity than NLI.

We utilized an existing corpus of Twitter conversations (Thought Vector, 2017) that consists of almost three million Tweets to and from the customer support accounts of 108 companies. We exemplarily chose the Tweets to and from AmazonHelp, AppleSupport, and SpotifyCares to ensure a comprehensive evaluation and preprocessed the dataset as follows. First, we filtered for conversations that involve exactly one company since only in these cases it is feasible to automatically decide which company is responsible for resolving the customer's inquiry. Second, we removed all conversations in a non-English language. For this purpose, we invoke the Python library "fastText" (Joulin et al., 2016, 2017), which provides language identification using a pre-trained model. Third, we deducted all Tweets that cannot be considered as conversations since the company did not reply to the customer's inquiry. Fourth, to improve the classification algorithm's accuracy, we applied spelling correction to all inbound Tweets using the Python library "pyspellchecker". Our final datasets for AmazonHelp, AppleSupport, and SpotifyCares consist of 288,828, 231,683, and 88,774 Tweets, respectively.

Following our approach in Figure 2, we first defined exemplary topics and process activities by analyzing samples of 200 inbound and 100 outbound Tweets. In the second step, we labeled them accordingly. To account for human errors in the labels, three of the authors checked and agreed on the labeled dataset. We labeled twice as many inbound Tweets as outbound Tweets since the inbound Tweets turned out to cover a broader range of topics, and in many cases, they could also stand alone without a response although the respective company answered them. Furthermore, our sample of inbound Tweets only contains Tweets that mark the beginning of the conversations since we observed that the customers usually describe their inquiry in the first message. The first decision point of the underlying support process seems to be of crucial interest from a process mining perspective (compared to decision points later on in the process). For example, depending on whether the customers describe an issue with their phone's battery or an issue with their computer's software, it is quite likely that different subprocesses handle their issues. Thus, we achieve to mine this central decision point reliably by labeling more inbound Tweets. After applying NLI with the default hypothesis "This example is [topic/activity]" in the third step, we investigated further hypotheses in the fourth step to improve the results for the topics and activities for which the default hypothesis did not yield satisfactory results. Following an iterative procedure, we included between 4 and 38 further combinations, such as "The sentence is about [topic/activity]", "The customer asks

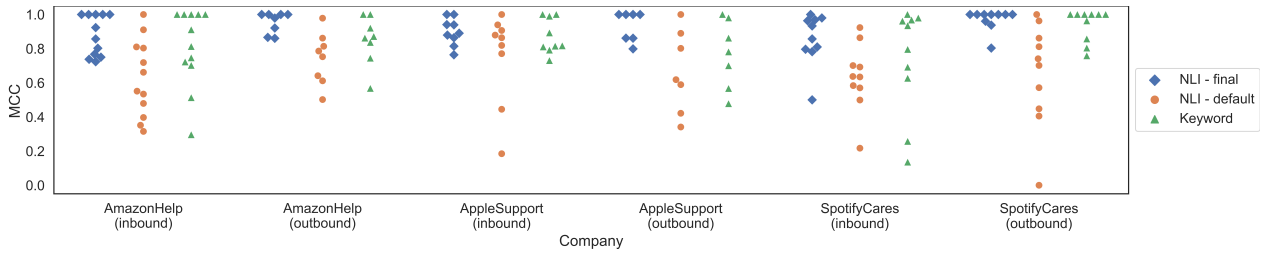


Figure 3: Classification Results for Inbound and Outbound Tweets of AmazonHelp, AppleSupport, and SpotifyCares

about [topic]", and "Please provide/send ...". Based on the results, we chose the NLI hypothesis with the highest MCC and the computed optimal threshold. In the last step, we constructed a standardized XES event log (IEEE, 2016) using "PM4Py" (Berti et al., 2023).

Figure 3 provides an overview of the results by plotting the MCCs of the three classifiers for the inbound and outbound Tweets of AmazonHelp, AppleSupport, and SpotifyCares. The blue diamonds, orange dots, and green triangles represent the distribution of the MCCs for classifying the topics (for inbound Tweets) and the activities (for outbound Tweets) using the "NLI - final", "NLI - default", and "Keyword" approach, respectively. The distributions lead us to the following conclusions: First, on average, the results achieved using the final NLI hypothesis outperform the other approaches. In 55 of 56 cases, the MCC is greater than 0.72, implying a high performance in all four dimensions of the binary classification confusion matrix. Therefore, we conclude a feasible approach for the construction of event logs. Second, in some cases, the "NLI - default" and, more particularly, the "Keyword" approach already achieve highly accurate results. For example, our labeled datasets indicate that customers having an issue with a particular product, name that product explicitly in their inquiry. Another example is when the customer support agent provides a URL to the customer. These URLs all start with "https://t.co" due to the Twitter URL shortening feature. Third, the "NLI - default" hypothesis ("This example is [...]") completed with the keyword has the highest variance in MCC among all approaches. We trace this observation back to some grammatically incorrect hypotheses. For example, "This example is what's happening" achieves an MCC of 0 on the outbound Tweets of Spotify, whereas the keyword approach using "what's happening" as well as the final hypothesis "This example asks to describe what's happening" achieves an MCC of 1.

To evaluate whether the constructed event logs are suitable for process mining purposes, such as process discovery, we imported the constructed log of AppleSupport again using "PM4Py" (Berti et al., 2023) and applied the Alpha Miner (van der Aalst et al., 2004), the Inductive Miner (Leemans et al., 2013), and the Heuristics Miner (Weijters et al., 2006). Before applying the miners, we filtered the event log for traces in which the customer's request could be assigned to one of our previously defined topics. Next, to reduce the vast variants, we exemplarily filtered the event log for the five most frequent variants. Figure 4 visualizes the resulting process maps portrayed as Petri nets of the three applied process discovery algorithms.

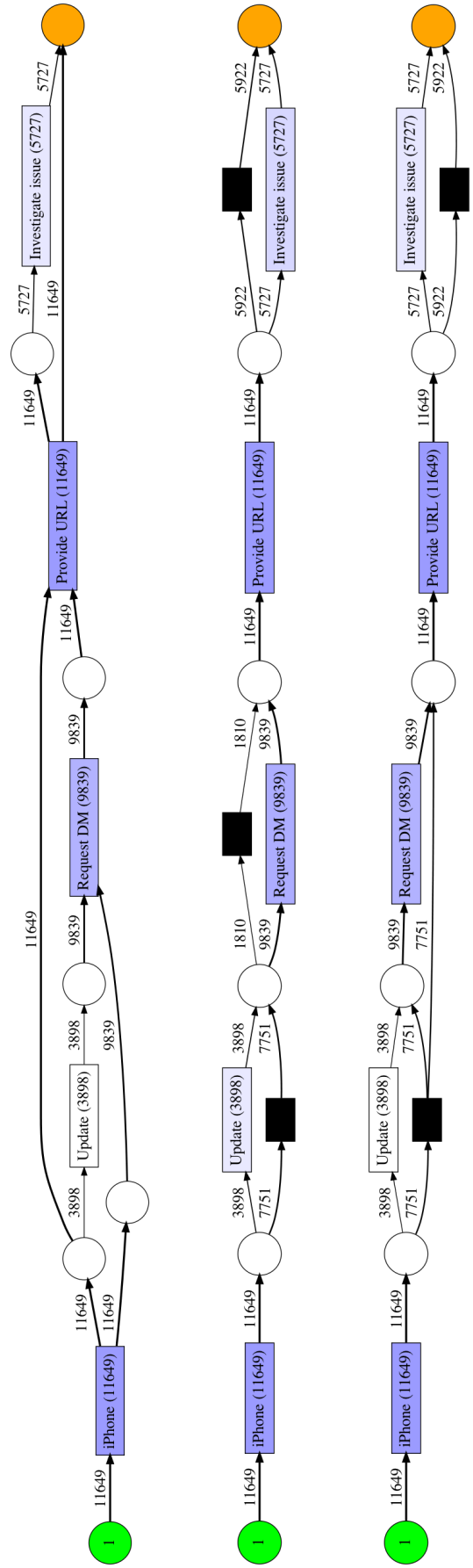


Figure 4: Process maps discovered by applying the Alpha Miner (top), Inductive Miner (center), and Heuristics Miner (bottom) of PM4Py to the five most frequent variants of the AppleSupport dataset

Although the Petri nets' visualizations differ among the algorithms, the Petri nets reflect the same process model. The models reveal that when a customer inquired about the topic "iPhone", the customer service agent provided a URL, regardless of whether the customer's issue could also be assigned to the topic "update". However, in 84 percent of cases, the agent asked the customer to send a direct message, and in 49 percent of cases, the agent further investigated the customer's issue.

In conclusion, RA 1 presents an approach to represent customer service conversations as a standardized IEEE XES event log that can be imported by common process mining applications. Our results show that NLI with a precisely formulated hypothesis about the topic or process activity of interest achieves the highest performance compared to two simpler baselines, benefiting from pretrained models' ability to understand natural language rather than merely memorizing co-occurring words. Although we demonstrated and evaluated the approach on written Twitter conversations, it generalizes to other channels, such as transcripts of customer service calls or internal communications. The contribution to research and practice is threefold. First, we show how NLI with a pretrained model enables the construction of high-quality event logs without requiring vast amounts of labeled data. Second, we demonstrate the applicability on real-world data from three companies and confirm that the constructed logs are suitable for process discovery. Third, we provide the full implementation and datasets on GitHub for reuse and extension.

2.2 Object-Centric Event Log Construction from Textual Data

While the NLI-based pipeline in Section 2.1 accurately produces case-centric event logs, real-world processes typically involve several interacting entities that a single-case view cannot capture (van der Aalst, 2023a). In recruitment, for example, one event may simultaneously concern multiple applicants, their individual applications, and a vacancy they all target. Flattening such multidimensional data into a single case representation leads to deficiency, convergence, and divergence problems (van der Aalst, 2019, 2023b).

Recognizing these limitations, the field has shifted from case-centric process mining to Object-Centric Process Mining (OCPM), which abandons the assumption that events belong to a single case and allows an event to reference multiple objects simultaneously (van der Aalst, 2023a; Wynn et al., 2022). This paradigm shift led to several OCEL formats, such as the OCEL 2.0 format, a comprehensive standard for OCPM, consisting of two main components: objects and events. Objects represent entities involved in processes, such as physical items (e. g., products or machines), abstract entities (e. g., orders or contracts), or individuals (e. g., employees or suppliers). Each object is categorized by an object type (e. g., product, order, or supplier), which defines specific attributes with dynamic values that can change over time. Object-to-Object (O2O) relationships represent structural or functional associations between objects, independent of specific events. Events in OCEL 2.0 represent discrete actions or occurrences within a process, such as approving an order, shipping an item, or making a payment. Each event is uniquely identified by an event ID and is characterized by an event type, which categorizes the nature of the event. Events are

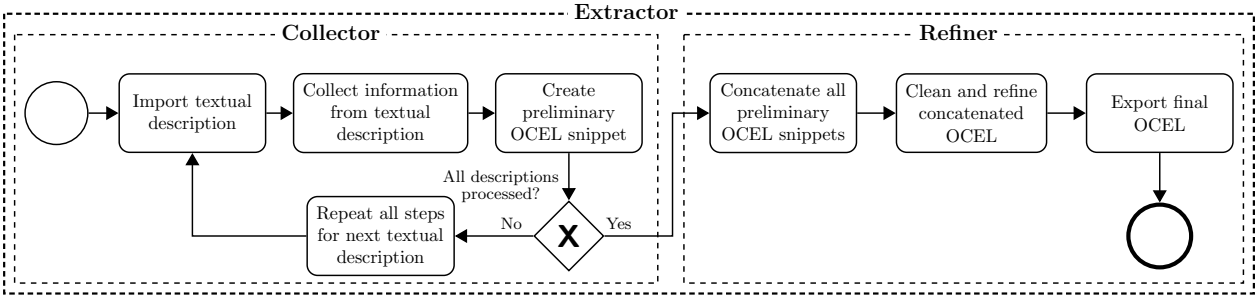


Figure 5: Pipeline for constructing OCELs from textual descriptions

atomic, meaning they occur at a specific point in time and do not span a duration. Each event type can also have additional attributes that provide further details about the event, with the flexibility to accommodate multiple values for these attributes. Furthermore, the OCEL 2.0 format captures relationships between events and objects through Event-to-Object (E2O) relationships to describe how objects participate in events and how events affect objects (Berti et al., 2024).

Although OCEL formats are now mature, to the best of our knowledge, there are no existing approaches that capture OCELs from unstructured textual descriptions. Building upon a prior conference publication (Buss et al., 2025), RA 2 (Buss et al., 2026) fills this gap by introducing an automated extraction approach using heuristic NLP and GenAI techniques. Following the Design Science Research (DSR) methodology of Peffers et al. (2007), we specify three design objectives: (1) event log extraction in an object-centric format, (2) event log extraction from unstructured textual descriptions, and (3) event log extraction at scale, fully automated, and without human intervention.

The functionality of our approach is illustrated in Figure 5. Initially, textual descriptions of arbitrary length are iteratively imported by the collector subcomponent. The collector aims to extract relevant information from each description and structure it into a preliminary OCEL format. Processing each description in isolation ensures that relevant details are captured without interference from other descriptions, making the approach more feasible for real-world applications. Working with smaller subsets reduces execution time and computational cost, creating a more efficient process. This incremental approach is also more realistic, as textual descriptions can be added progressively instead of requiring all data to be gathered upfront.

Once these preliminary OCEL snippets are gathered, they are passed to the refiner subcomponent, whose goal is to ensure the coherence and quality of the final output. Therefore, the refiner concatenates the individual snippets into a unified version, which it then cleans, refines, and harmonizes. This step is necessary because, without refinement, inconsistencies or redundant information across descriptions could lead to misinterpretations or an incomplete representation of the process. The refiner aligns data structures and terminology across entries, enhances data quality by removing noise or errors, and ensures compatibility of all elements. This process ultimately improves the precision of the final OCEL, making it a more reliable and valuable resource for further analysis and decision-making.

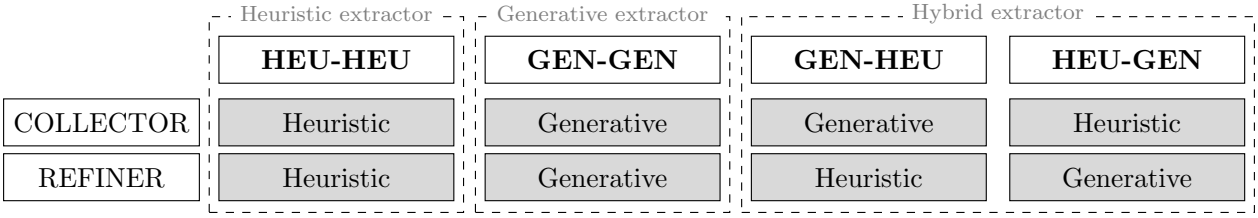


Figure 6: Four extractor variants

We instantiate our approach in four distinct extractor variants, leveraging heuristic NLP and GenAI techniques, to evaluate their respective strengths and weaknesses. The first extractor variant corresponds to a heuristic extractor (HEU-HEU) that features a heuristic collector and a heuristic refiner. The second variant is a generative extractor (GEN-GEN) consisting of a generative collector and a generative refiner. Lastly, we also develop two hybrid extractors, namely a GEN-HEU extractor, employing a generative collector and a heuristic refiner, and a HEU-GEN extractor, consisting of a heuristic collector and a generative refiner. Figure 6 illustrates the four extractor variants.

The four extractor variants are developed sequentially within our development framework that enables continuous validation of the extraction capabilities of the different extractor variants. Figure 7 illustrates the three components of the framework: a generator instance, an extractor instance, and a comparison instance.

Initially, a test subset consisting of an original OCEL is provided to the *generator instance* of the development framework. The generator instance is then tasked with converting the events of the OCEL into corresponding textual descriptions. To ensure the comparability of the results with our prior conference publication (Buss et al., 2025), we use the same textual descriptions (generated with OpenAI's `gpt-4o-mini-2024-07-18` LLM) in RA 2.

The generated synthetic textual descriptions are then handed over to the *extractor instance* of the development framework. The extractor instance, corresponding to our four extractor variants, is then tasked with analyzing the provided textual descriptions to reconstruct the original OCEL. Each extractor variant, therefore, leverages its respective heuristic and generative subcomponents. As a result, one extracted OCEL is created for each original OCEL.

Lastly, the extracted OCELs are compared with their original counterparts using the *comparison instance*, which evaluates the alignment of the logs across various categories and levels of detail. The levels of detail, comprising parent levels and absolute and relative child levels, follow the structure of the OCEL 2.0 format. At the parent level, categories such as object types, event types, object instances, and event instances are analyzed to ensure the existence of corresponding values in the extracted logs. Meanwhile, the child levels assess whether specific child values are accurately mapped to their parent categories. For example, the comparison verifies whether object types, attribute types, and attribute values are correctly linked to object labels, and whether the appropriate O2O and E2O relationships are identified. This analysis is performed both on an absolute level (across the entire event log) and a relative level (assuming the parent category is correctly

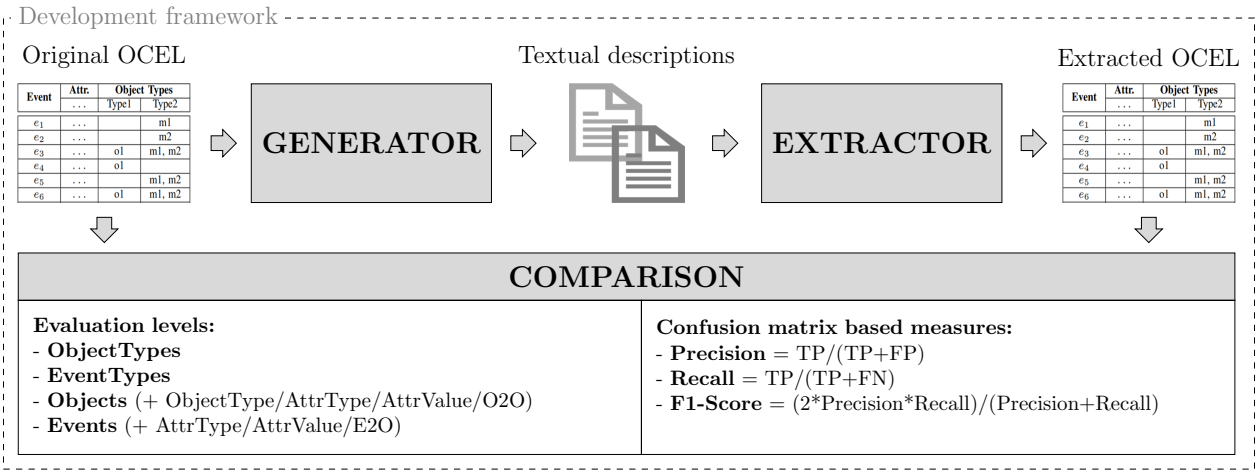


Figure 7: Development framework

identified). To measure the correctly identified values, we calculate precision, recall, and F1-score. Precision measures the proportion of correct identifications among retrieved entities, while recall quantifies how many relevant entities from the original dataset were successfully retrieved. Since precision and recall often trade off against each other, the F1-score balances these metrics by calculating their harmonic mean (Derczynski, 2016). To report the overall performance on a particular dataset, we calculate the overall precision and recall by averaging the precision and recall across all parent levels and absolute child levels, and the overall F1-score as the harmonic mean of the resulting overall precision and recall values.

Following the generic approach shown in Figure 5, textual descriptions are iteratively provided to the *heuristic collector*, which constructs preliminary OCEL snippets for the corresponding descriptions. Therefore, the heuristic collector first processes the provided textual description using a SpaCy NLP parsing pipeline (Vasiliev, 2020). This pipeline tokenizes the text and extracts key token features, including dependency labels, Part-of-Speech (PoS) tags, Named Entity Recognition (NER) labels, and syntactic dependency relations such as children and ancestor tokens. Following a set of predefined rules, the heuristic collector evaluates the tokens, their dependencies, PoS tags, and NER tags to identify candidate values for the essential OCEL components: timestamps, activities, object labels, object types, attribute values, and attribute types. After refining the extracted values through lemmatization, analysis of their surroundings for reference values, and filtering redundant words extracted for multiple categories, they are mapped to each other according to defined relationships. The mappings include object labels to object types, attribute values to attribute types, object labels to other object labels to reveal O2O relationships, activities to timestamps, attributes to timestamps, object labels to activity-timestamp combinations to extract E2O relationships, and attribute values to object labels and activity-timestamp combinations. Based on these mappings, the heuristic collector generates a preliminary OCEL snippet per textual description. For this instantiation, we decided to export the OCEL snippets into the OCEL 2.0 format.

The *heuristic refiner* first concatenates preliminary OCEL snippets into a unified log. The unified log then undergoes a series of cleaning and refinement steps, leveraging heuristic rules and majority-based approaches, that are repeated until the log attains a final state with a maximum of five iterations. Within these iterations, the refiner alleviates data quality issues by, for example, resolving name inconsistencies, merging synonyms, and enforcing alignment between the object types, event types, objects, and events sections of the OCEL.

The *generative collector* is implemented using OpenAI's `gpt-5-mini-2025-08-07` LLM with high reasoning effort provided through the Azure OpenAI Application Programming Interface (API). Whereas the implementation in our earlier conference publication (Buss et al., 2025) utilized `gpt-4o-mini-2024-07-18` and the assistants API with file search capabilities, the current implementation solely relies on the chat completion API. After importing a textual description, the LLM is invoked with a system prompt containing instructions on the desired output and a user prompt containing the textual description. We formulate all prompts using advanced prompting techniques (Schulhoff et al., 2025). The system prompt starts with assigning a role to the LLM, instructing it to act as a "process mining expert" extracting OCELs from textual descriptions, followed by an empty OCEL. Next, the task is further specified by explaining the OCEL components to extract, along with detailed descriptions taken from the OCEL 2.0 specification (Berti et al., 2024). Furthermore, we explicitly instruct the LLM to solely use information from the provided textual descriptions and how to represent E2O and O2O relationships. Finally, we specify the OCEL 2.0 standard as the desired output format. In addition to a corresponding instruction in the prompt, we set the `response_format` parameter available through the Azure OpenAI API to `json_schema` and provide the JavaScript Object Notation (JSON) schema from the OCEL 2.0 specification (Berti et al., 2024) as the desired schema. Following this approach, we iteratively provide the textual descriptions to obtain a preliminary OCEL snippet per textual description.

The *generative refiner* combines the preliminary OCEL snippets into a concatenated OCEL based on predefined rules to ensure that the concatenated event log adheres to the OCEL 2.0 standard. The subsequent refinement is implemented as a sequence of LLM-guided normalization steps utilizing OpenAI's `gpt-5-mini-2025-08-07` with strict JSON output control and additional deterministic post-processing. In contrast to the implementation in our earlier conference publication (Buss et al., 2025), which just submitted the whole concatenated OCEL along with a respective refinement prompt to `gpt-4o-mini-2024-07-18`, the present implementation significantly increases the scalability of our approach to larger volumes of unstructured data for two reasons. First, the refiner modularizes LLM prompts by OCEL component, which makes the LLM's input length limit far harder to reach than with the entire OCEL. Second, responses are constrained by a JSON schema and applied through deterministic rewrite and validation rules, which improves internal consistency across OCEL components and conformance with the OCEL 2.0 JSON format. Several steps are executed in parallel to further reduce runtime on larger OCELs.

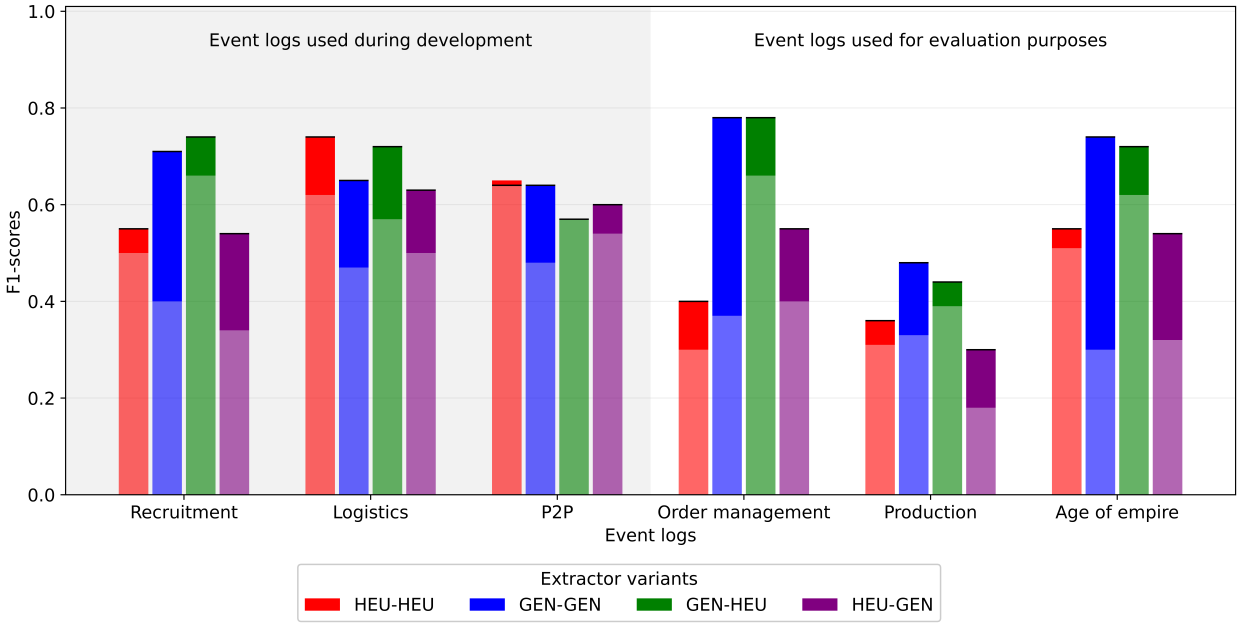


Figure 8: F1-score by event log and extractor variant. Saturated overlays indicate the change in performance compared to our previous conference publication (Buss et al., 2025)

To comprehensively assess the extraction capabilities of the four instantiated extractor variants, we create synthetic textual descriptions for six publicly available event logs in the OCEL 2.0 format, provide these descriptions to the four extractor variants, and compare the reconstructed logs with their original counterparts. Three of these logs – the recruitment log (Berti, 2023), logistics log (Knopp and Graves, 2023), and the Procure-To-Payment (P2P) log (Park and Tacke genannt Unterberg, 2023) – were previously employed in the development and validation of the heuristic extractor. The remaining three logs – an order management log (Knopp and van der Aalst, 2023), a production log (Heinisch et al., 2024), and an Age of Empires log (Liss et al., 2024) – were not used during development, providing an opportunity to evaluate the generalization capabilities of each extractor variant. A standardized test subset of 1,000 events is created for each event log. The test subsets are then processed by the generator instance of the development framework, tasked with converting the events into textual descriptions across three levels of complexity. The synthetic textual descriptions are then provided to the four extractor variants. These variants leverage their respective heuristic and generative collector and refiner subcomponents to reconstruct the original logs. Finally, the extracted OCELs are compared with their original counterparts using the comparison instance of the development framework.

Figure 8 shows the F1-score per event log and for each of the four extractor variants, including the change in performance compared to our previous conference publication (Buss et al., 2025). In four event logs (recruitment, order management, production, and Age of Empires), the GEN-GEN and GEN-HEU extractor variants achieve comparable performance and significantly outperform the other two variants. In contrast, all four variants achieve comparable performance on the logistics and the P2P event log.

Compared to our previous conference publication (Buss et al., 2025), we observe a significant increase in performance of the GEN-GEN extractor, achieving an improvement of 0.28 in F1-score across the six event logs. We attribute this improvement primarily to the revised implementation of both generative components (as described above) and to updating the underlying LLM from `gpt-4o-mini-2024-07-18` to `gpt-5-mini-2025-08-07`. As the increase in performance of the GEN-HEU and the HEU-GEN variant shows (an increase of 0.08 and 0.15, respectively), the changes in both generative components contributed to the overall improvement. Although the implementation of the heuristic components did not change, we observe an overall performance increase of 0.06 (despite the decline by 0.01 on the P2P event log) in the respective F1-score. We trace this observation back to changing the utilized SpaCy language model from small (`en_core_web_sm`) to medium (`en_core_web_md`) and a minor fix in the comparison instance to improve the matching of original and retrieved components. Nevertheless, the different performance of the HEU-HEU extractor variant on the three event logs used during development, compared to the performance on the three event logs for evaluation purposes, indicates that the heuristic extractor was fine-tuned to the characteristics of the former event logs. In conclusion, both the GEN-GEN and the GEN-HEU extractor variants outperform the other variants, particularly on event logs not used during the development of the heuristic components. Therefore, we conduct another evaluation on naturally occurring textual descriptions and examine how well the two best-performing variants generalize beyond the synthetic datasets.

For this purpose, we select fire status updates regularly posted by the California Department of Forestry and Fire Protection[1], which provide real-time information about ongoing wildfire events. Each status update is transformed into a separate textual description, capturing essential details such as the specific fire involved, the date and time the update was reported, and the situation summary provided in the original status update. As a result, we collected 280 status updates, distributed across the 10 fire incidents.

The textual descriptions are then processed in isolation by both the GEN-GEN and the GEN-HEU extractor variants, which are tasked with creating an OCEL based on the provided fire status updates. The GEN-GEN extractor variant identifies 306 object instances across 27 object types and 628 event instances belonging to 199 event types. In turn, the GEN-HEU extractor variant identifies 381 object instances across 52 object types and 547 event instances across 103 event types. Both extractor variants identify a similar number of E2O relationships (GEN-GEN: 2185, GEN-HEU: 1843).

Using the `discover_ocdfg` function provided by the Python library PM4Py (Berti et al., 2023), we derive Object-Centric Directly-Follows Graphs (OCDFGs) to visualize the efforts taken for the different fire incidents. Figures 9 and 10, for instance, showcase the OCDFG with frequency annotations for a specific fire instance, namely the Lilac Fire that started on January 21, 2025, in San Diego. Both figures were created with an activity and edge threshold of 2, thus containing events and transitions for the same object type occurring at least twice in the corresponding OCEL.

[1] `https://www.fire.ca.gov/incidents/`

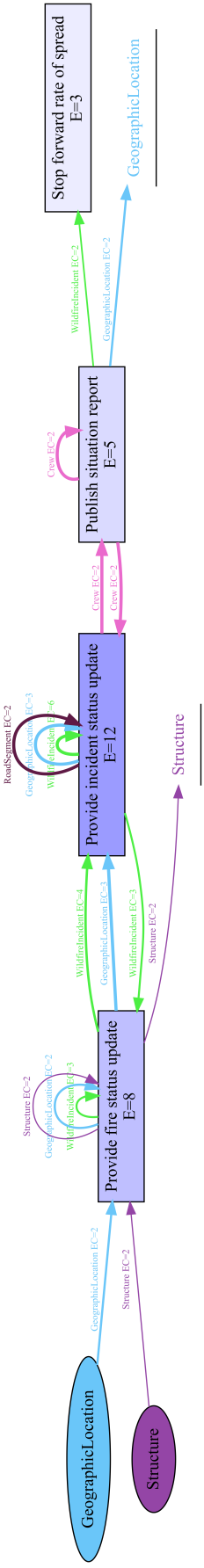


Figure 9: OCDFG based on an OCEL derived with the GEN-GEN extractor variant for fire status updates regarding the Lilac Fire in January 2025 in San Diego

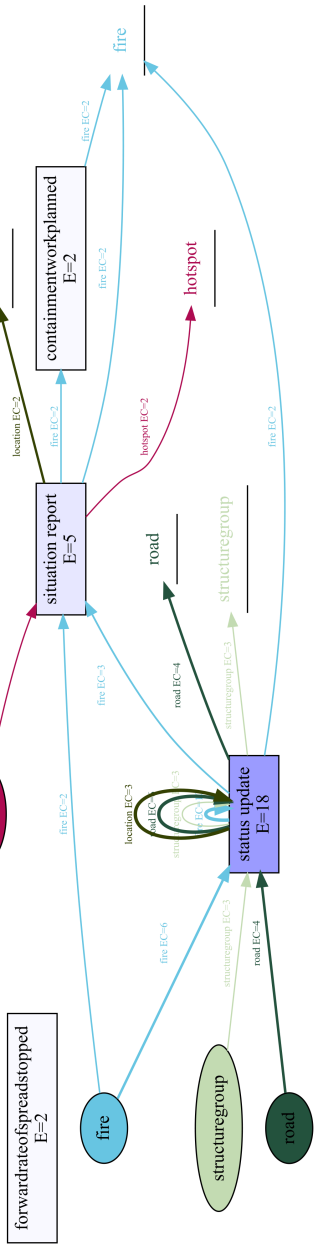


Figure 10: OCDFG based on an OCEL derived with the GEN-HEU extractor variant for fire status updates regarding the Lilac Fire in January 2025 in San Diego

Figure 9 (based on the GEN-GEN extractor variant) presents a compact control flow with four event types, three of them related to providing an update on the Lilac Fire and the fourth indicating progress in the containment efforts. The high number of events per event type indicates that multiple updates are published for a single fire, with the update frequently containing the structure of the fire, the involved firefighter crew, and geographical information, such as the location of the fire. In contrast, Figure 10 (based on the GEN-HEU extractor variant) explicitly contains an event type modeling the planning of containment work and exposes a slightly richer set of object types. However, the event type and object type vocabularies are less normalized (e. g., concatenated labels such as `forwardrateofspreadstopped`), thus impeding the readability and interpretability in practice.

Taken together, the results on naturally occurring textual descriptions show that both the GEN-GEN and GEN-HEU extractor variants generalize beyond the synthetic datasets. Compared side by side, the GEN-GEN extractor variant delivers a more coherent and interpretable representation while preserving coverage, as it uses 48% fewer object types than GEN-HEU (27 vs. 52) and still extracts about 16% more event instances (628 vs. 547). The density of E2O relationships remains comparable, but the resulting OCDFG from GEN-GEN forms a single connected component. Based on these findings, we recommend GEN-GEN as the default extractor for real-world textual descriptions. Its clearer labels and more cohesive graph reduce analyst effort without sacrificing the richness needed for decision-making in practical application contexts.

In summary, RA 2 proposes a novel approach for automatically extracting OCELs from unstructured textual descriptions using heuristic NLP and GenAI techniques. Following the DSR methodology, we considered three design objectives in the development of our approach, which comprises two key subcomponents: a collector that identifies events and objects (including their attributes and relationships), and a refiner that consolidates and cleans the extracted information. Both subcomponents were instantiated in heuristic and generative forms, resulting in four distinct extractor variants. The results showed that both the fully generative configuration (GEN-GEN) and the hybrid GEN-HEU extractor variant achieved the highest overall extraction performance and outperformed the fully heuristic (HEU-HEU) and the opposite hybrid extractor variant (HEU-GEN). Considering that the GEN-GEN extractor variant exhibits the highest levels of extraction quality, generalization capabilities, and semantic utility, and furthermore enables the creation of coherent and interpretable visualizations, we recommend GEN-GEN as the default extractor for real-world textual descriptions. The contribution of RA 2 is threefold. First, it presents a novel approach for automatically extracting OCELs directly from unstructured textual descriptions, filling a notable gap in process mining research by enabling the incorporation of textual data that captures complex, object-centric process behavior. Second, implementing and evaluating four distinct extractor variants offers a systematic comparative analysis of heuristic NLPs and GenAI techniques in this context. Third, to foster reproducibility and future research, we publish our implementation and datasets publicly available on GitHub.

3 AI-Driven Process Analysis and Redesign

Having established transparent insight into the current state of a business process and derived corresponding as-is process models in Section 2, the BPM lifecycle advances to the analysis phase, providing insights on weaknesses and their impact, and the redesign phase, yielding a better-performing to-be process model. While process mining has matured into a powerful, data-driven means to diagnose process inefficiencies, the subsequent steps of turning diagnosed issues into well-founded improvement ideas and viable to-be models remain heavily manual, creativity-intensive, and resource-demanding (Figl and Recker, 2016; Groß et al., 2024; Gross et al., 2019; Huang et al., 2015; Mustansir et al., 2022; Reijers and Liman Mansar, 2005). Extant studies consistently report that BPI initiatives consume scarce expert time, require deep domain knowledge, and hinge on the generation and rigorous evaluation of redesign alternatives (Bader et al., 2023; Beerepoot et al., 2019; Zellner, 2011).

Recent calls for PIISs (Fehrer et al., 2025; Moder et al., 2025) urge the community to provide (semi-)automated, scalable support for BPI initiatives (Beerepoot et al., 2023; Park and van der Aalst, 2022). However, current computational approaches typically automate only isolated activities (e. g., recommending redesign patterns or applying individual optimization heuristics) and thus fall short on one or more essential BPI capabilities. For example, these approaches struggle to consider existing BPI knowledge (e. g., Afflerbach et al., 2017), to span a solution space beyond patterns or boundaries imposed by process input data (e. g., Fehrer et al., 2022; Niedermann and Schwarz, 2011), to ensure feasibility of the redesigned process models (e. g., van Dun et al., 2023), or to assess the improved process models quantitatively (e. g., Truong and Lê, 2016).

At the same time, advances in GenAI, most notably LLMs, open up new opportunities (Feuerriegel et al., 2024; Vaswani et al., 2017). LLMs combine computational creativity (Zhao et al., 2025) with strong reasoning capabilities (Kojima et al., 2022), but their direct use in BPI is constrained by well-known issues such as hallucinations and limited controllability of outputs (Agrawal et al., 2024; Huang et al., 2025; Maynez et al., 2020; Ye et al., 2024). RAG addresses these weaknesses by grounding LLMs in external, continuously updateable knowledge bases, thereby improving factuality and contextual relevance (Balaguer et al., 2024; Lewis et al., 2020b; Shuster et al., 2021). Beyond ideation, quantitative assessment remains indispensable: improved process models must be sound, feasible, and outperform the as-is process models on performance metrics, such as cycle time and execution cost (Dumas et al., 2018).

Against this background, this section presents two complementary artifacts that support AI-driven process analysis and redesign. Both artifacts are developed with the DSR paradigm (Gregor and Hevner, 2013; Peffers et al., 2007). The first artifact, the *Process Improvement Copilot* presented in Section 3.1 (RA 3), a RAG-enhanced LLM-based PIIS, bridges the gap between identified process inefficiencies and actionable, context-specific improvement ideas. It draws on process execution data in the form of event logs and existing BPI knowledge (BPI patterns and case studies) to generate justified process improvement ideas at scale, thus reducing the dependence on scarce expertise.

The second artifact, *ABuPrOpt*, presented in Section 3.2 (RA 4), uses LLMs together with simulation to generate and quantitatively assess improved, sound, and feasible process models. Designed along three explicit design objectives, ABuPrOpt exhibits an elevated solution space, incorporates standard and custom improvement objectives, and quantifies the cycle time and execution cost of the suggested process models. Together, these studies demonstrate how GenAI in the analysis and redesign phases of the BPM lifecycle can not only uncover weaknesses in business processes, but also generate and assess ideas on how the business process at hand should be changed.

3.1 Retrieval-Augmented Generation of Context-Specific Process Improvement Ideas

Once process inefficiencies have been identified, organizations still struggle to come up with targeted improvement ideas. Traditionally, generating BPI ideas involves domain experts (Mustansir et al., 2022) and relies on manual methods and techniques, such as brainstorming (Kettinger et al., 1997) and exploring best practices collected from successful BPI projects (Reijers and Liman Mansar, 2005). These best practices are called BPI patterns, enable time and effort saving (as BPI patterns are proven solutions that might be reused in similar contexts), and simplify the process for novices (Falk et al., 2013). However, manual ideation remains slow and expertise-intensive, existing PIISs seldom reuse the rich body of BPI knowledge systematically, and the matching between process inefficiencies and process improvement ideas is done without consideration of the context. Moreover, BPI initiatives often involve multiple inefficiencies at once and thus require coherent idea sets rather than isolated fixes (Li et al., 2023; Tang et al., 2023).

To address these challenges, RA 3 (Smalei et al., 2026) presents the *Process Improvement Copilot*, a RAG-enhanced LLM-based PIIS that supports the generation of process improvement ideas for previously identified process inefficiencies. The artifact – developed in the spirit of the DSR paradigm (Peffers et al., 2007) – leverages existing BPI knowledge (in the form of BPI patterns and case studies), process execution data, and process context (vom Brocke et al., 2016) to automatically generate process improvement ideas, thus reducing the time and BPI expertise required for this step. Additionally, the Process Improvement Copilot can handle complex BPI initiatives with multiple process inefficiencies. The proposed PIIS facilitates follow-up actions by justifying each idea and showing the respective pieces of knowledge used to generate each idea.

The Process Improvement Copilot incorporates five design objectives in its system architecture: First, it leverages accumulated BPI knowledge by incorporating a collection of BPI patterns and case studies. Second, it generates context-relevant process improvement ideas by utilizing event logs and the process context specified by the users. Third, it reduces the time and BPI expertise required to carry out the idea generation by employing a computational algorithm to automate idea generation. Fourth, it handles complex BPI initiatives by consolidating all generated ideas

in a coordinated manner. Fifth, it facilitates human-on-the-loop follow-up actions by showing the knowledge chunks used for idea generation and thorough prompt engineering and LLM configuration to minimize hallucinations.

Figure 11 depicts the system architecture of the Process Improvement Copilot. It can be divided into four major components. The *Inefficiency Finder* analyzes process data ingested from the event log using one of the *Inefficiency Pattern Finders* (we use three patterns for demonstration purposes: *Activity Variants*, *Frequent Handovers*, and *Rework*). Each *Inefficiency Pattern Finder* generates a list of identified process inefficiencies as an output.

The *Retriever* is responsible for accessing and selecting relevant BPI knowledge. It operates on a curated *Collection of BPI Knowledge* encompassing BPI patterns and case studies. To enable efficient retrieval, the BPI knowledge is pre-processed by a *Chunking Model* and an *Embedding Model*, resulting in a *Vector Database* of vectorized knowledge chunks. Upon receiving a query (list of identified process inefficiencies), the *Retriever* performs a *Similarity Search* between process inefficiency and knowledge chunks to extract a *List of Relevant Knowledge Chunks*. The *Retriever* component provides the relevant BPI knowledge for further idea generation.

The *Idea Generator* takes four data entities as input (*Process Context*, *List of Process Inefficiencies*, *List of Relevant Knowledge Chunks*, and *Idea Generation Prompt*) and invokes the *LLM for Idea Generation* element to generate a *List of Process Improvement Ideas*. The LLM element leverages retrieved knowledge and contextual information to generate relevant process improvement ideas with justifications.

The *Idea Combinator* takes the *List of Process Improvement Ideas* generated by the *Idea Generator* as input. Another LLM (*LLM for Idea Combination*), guided by an *Idea Combination Prompt*, is employed to consolidate and potentially enhance these initial ideas. The aim is to produce a *Consolidated List of Process Improvement Ideas* ready for the user review.

There are two additional elements that are not part of any components. These are *Manual Inefficiency Input* and *Process Context Input*. Using the *Manual Inefficiency Input* element, the user can manually submit any process inefficiency to the *List of Process Inefficiencies*. With the help of the *Process Context Input* element, the user can provide the system with a comprehensive context description. We structure the input about context along contextual factors proposed by vom Brocke et al. (2016): goal dimension, process dimension, organization dimension, and environment dimension.

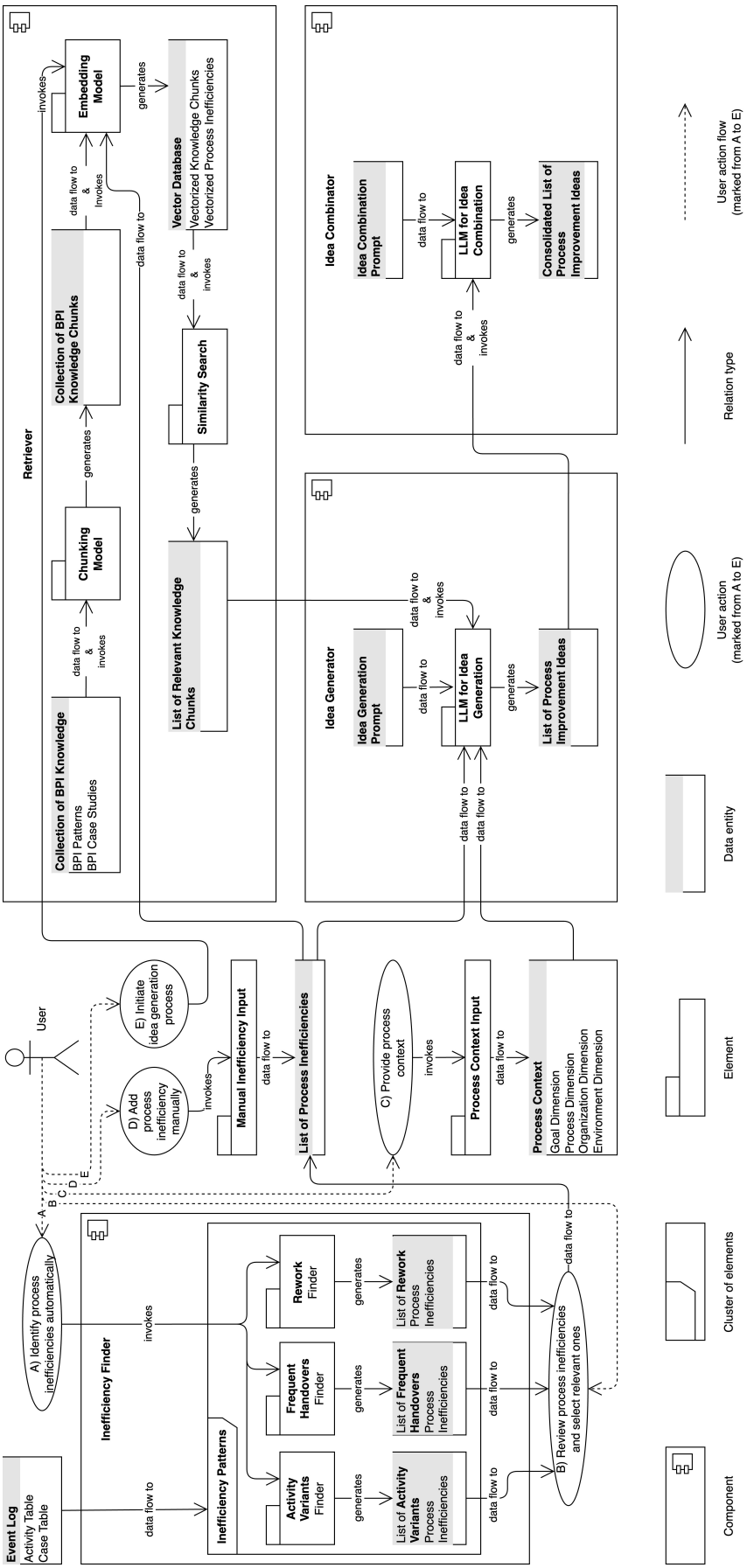


Figure 11: System architecture of the Process Improvement Copilot

We evaluated the technical feasibility of the proposed system architecture through a minimum viable product in Python. To this end, we formulated several hypothetical process inefficiencies (e. g., "Activity 'Review the document' is only executed once per week. It is executed by a person from a higher level of hierarchy. Until then, all cases are blocked.") and set up a basic RAG pipeline operating on a sample of seven BPI patterns. Internal testing indicated that the approach is technically feasible: the RAG-based prototype consistently selected relevant BPI patterns and produced process improvement ideas that were both higher in quality and more diverse than ideas generated by a plain LLM relying solely on its parametric memory. Moreover, the logical flow from process inefficiency detection to selecting relevant pieces of knowledge and finally to idea generation mirrored human reasoning, which we found beneficial for user understanding of the overall concept.

Building on this foundation, we implemented the system architecture as a full prototype. The prototype ingests event logs, applies three selected inefficiency patterns (*Activity Variants*, *Frequent Handovers*, and *Rework*) adapted from Lashkevich et al. (2023) to automatically derive textual descriptions of inefficiencies, retrieves matching knowledge chunks from a curated base of BPI patterns and case studies via *Similarity Search* in the *Vector Database*, and invokes the *LLM for Idea Generation* to generate specific process improvement ideas. When multiple inefficiencies are present, the prototype employs the *LLM for Idea Combination* to refine the initially generated ideas into a *Consolidated List of Process Improvement Ideas*. Figure 12 illustrates the resulting output format, showing how each idea is accompanied by its justification and the specific knowledge pieces retrieved.

The user interaction is supported through an interface that exposes all necessary actions: providing or adjusting process context, optionally entering inefficiencies manually, triggering idea generation and combination, and reviewing the consolidated list of improvement ideas. Prompts guiding both LLMs follow advanced prompting techniques (Schulhoff et al., 2025): the model is assigned a role as an "experienced assistant specializing in business process improvement", its objective, requirements, and constraints are specified, and a structured output format is requested. To prioritize determinism over creativity, both LLMs operate with a temperature parameter set to 0, ensuring reproducible outputs while still leveraging the knowledge base.

To comprehensively evaluate the design objectives, the system architecture, and the software prototype, we conducted 13 semi-structured interviews with 16 experts from academia and industry. We chose this method due to its flexibility regarding possible question adjustments based on the interviewee's experience and competence (Magaldi and Berler, 2020). The interviewed academic experts are doctoral candidates and researchers with completed doctoral degrees in the field of information systems from various German research institutions, whereas the industry experts are employees of large corporations, mid-size companies, and start-ups from different industries. Most of the industry experts are responsible for BPI initiatives in their organizations.

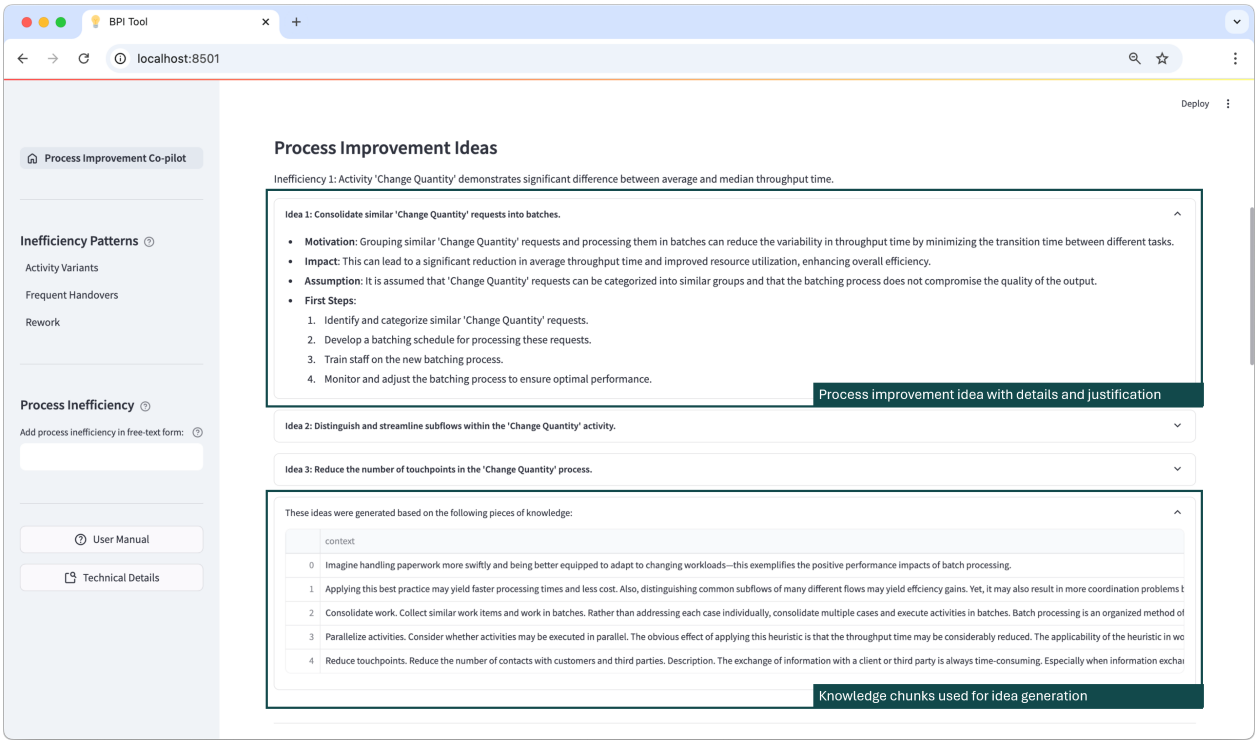


Figure 12: Output of the Idea Generator component

During the interviews, we used an online form to facilitate the collection of quantitative feedback on a 5-point Likert scale (Joshi et al., 2015) and to gather demographic information on the participants. The interviews averaged 54 minutes and always started with (1) an introduction round, followed by (2) a presentation of the research motivation, (3) an evaluation of the importance of the problem area and the design objectives, (4) an introduction of the system architecture, (5) a demonstration and evaluation of the software prototype instantiated on an academic purchase-to-pay event log, (6) and general feedback on the approach and results. We defined each of the evaluated terms to guarantee a shared understanding among all interviewees.

The experts emphasized the critical role of BPI in organizational success and agreed with the existing gap in computational support of the idea generation stage. 9 out of 16 experts ranked the identified problem area as extremely important and 5 out of 16 as very important and emphasized that improving business processes allows organizations to allocate resources more effectively, optimize time utilization, and enhance customer value. Furthermore, the experts expressed optimism about the potential for computational support in BPI. They recognized the potential for a technological solution to lower the mental hurdle for idea generation, making it easier for people to initiate actions and implement changes. Some experts noted the potential of computational support to bring structure and organization to the steps of the BPI process. At the same time, many experts acknowledged the difficulty of providing specific context-driven ideas. They expressed the concern that the intricacies of domain understanding, process context, BPI expertise, and common sense are too complex for a comprehensive PIIS.

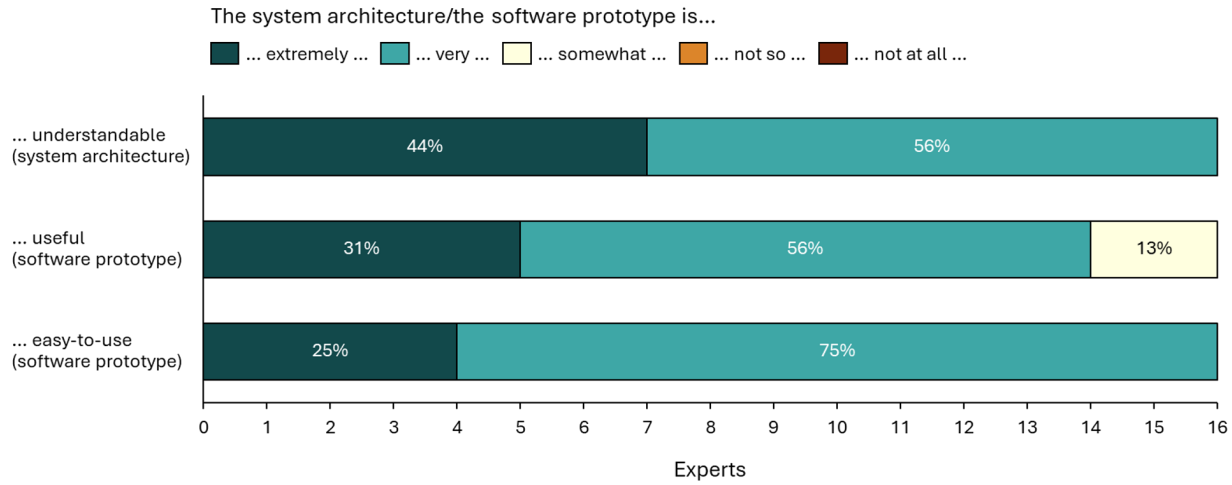


Figure 13: Evaluated understandability, usefulness, and ease-of-use of the system architecture and the software prototype

We evaluated the system architecture with regard to its understandability and the software prototype with regard to its usefulness and ease of use (Davis, 1989; Sonnenberg and vom Brocke, 2012b). In the interviews, we provided the following definitions of the terms. *Understandability* is how well the system architecture is to understand, making it easy for others to replicate and build upon. *Usefulness* is the degree to which the software prototype effectively solves a problem and adds value to real-world situations. *Ease-of-use* is the degree of user-friendliness and simplicity in interacting with the software prototype, making its effective utilization straightforward for individuals.

The interviewees pointed out a high level of *understandability* of the system architecture (see Figure 13). Even without extensive previous experience with RAG, the experts were able to grasp the whole concept. They confirmed that, from a logical perspective, the selected approach to generating process improvement ideas for process inefficiencies resembles the human way of thinking and, hence, is suitable. To account for different levels of the user's experience with the technology, one interviewee suggested explaining the concept at different levels of abstraction: detailed technical explanation with implementation details for experts as well as high-level conceptual explanation with real-world analogies for non-experts.

Regarding the *usefulness* of the software prototype, we observed two groups of experts. The former group clearly distinguished the *usefulness* of the software prototype from the *usefulness* of the generated process improvement ideas. They pointed out that the solution leverages accumulated knowledge about BPI, enabling users without extensive BPI expertise to commence BPI initiatives and additionally leading to the reduction of the needed time. The latter group emphasized the inseparability of the *usefulness* of the software prototype from the *usefulness* of the generated process improvement ideas in their opinion.

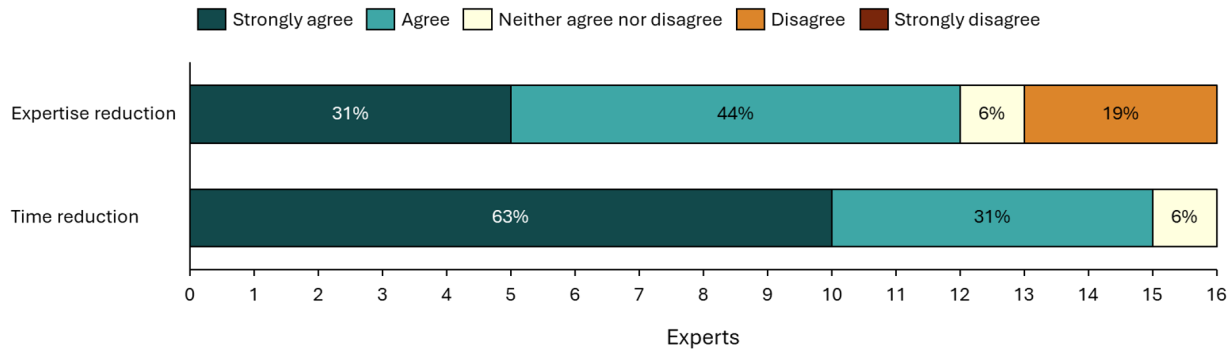


Figure 14: Evaluated expertise reduction and time reduction for the idea generation process attributable to artifact use

The majority of the experts ranked the software prototype as very *easy-to-use*, highlighting potential for improvement. They appreciated a lean, uncluttered interface with minimal distractions, allowing users to focus on the task at hand. However, in most of the interviews, the experts shared some ideas about improving the *ease-of-use* (e. g., enhanced user guidance, more detailed explanations and examples, and clearer structure and navigation).

In line with the third design objective, we evaluated the software prototype's impact on the expertise and time required to conduct BPI initiatives. While acknowledging significant time savings, the experts expressed varying opinions on the reduction of expertise required (see Figure 14). 10 out of 16 experts expressed that they strongly agree that the software prototype reduces the time needed to generate process improvement ideas and justifications. The opinions on the reduction of required expertise were more varied. While confirming that the software prototype significantly reduces the need for specialized knowledge, many experts simultaneously emphasized that process-specific expertise remains crucial.

The experts furthermore emphasized that the software prototype's lean design and straightforward user interface significantly simplifies the user interaction and streamlines the focus on the core functionality – the generation of process improvement ideas. Additionally, the experts provided specific suggestions for the further development of the prototype.

To validate the proof of value of the Process Improvement Copilot in a real-world setting, we conducted a process improvement workshop at a multinational technology conglomerate specializing in digitalization and automation. The 90-minute workshop involved eight participants from different departments and seniority levels, and was separated into two parts. In the first part, after an introduction to the problem area and research motivation, we showed the participants the system architecture and demonstrated the software prototype to collect feedback on their *understandability*, *usefulness*, and *ease-of-use*. In the second part, we replicated the setting of a process improvement workshop and instantiated the software prototype on the purchase-to-pay event log of the conglomerate's supply chain management department. The participants identified inefficiencies and generated improvement ideas, and then evaluated the ideas produced by the software prototype.

At the end of part one, the system architecture's *understandability* and the software prototype's *usefulness* and *ease-of-use* were rated using the same definitions as in the interviews. The ratings were slightly lower than in the expert interviews across all three criteria, which we trace back to the workshop setting (eight participants together, potentially inhibiting individual questions compared to the interview setting with one or two participants where clarifying questions could be asked more freely) and differing expectations (while the expert interviews were primarily focused on the underlying concept, the workshop participants envisioned the application of the Process Improvement Copilot in their daily operational contexts and might have set a different baseline).

In part two, the workshop participants were able to automatically identify process inefficiencies or manually submit their own ones and automatically generate process improvement ideas using the Process Improvement Copilot. Each of the eight participants ranked six generated process improvement ideas, resulting in 48 rankings. The results indicate that the majority of ideas fell into the *neither agree nor disagree category* for both the relevance of the idea and the quality of follow-up recommendations (60% and 52% of the ideas, respectively), whereas approximately 20% of the generated process improvement ideas were considered relevant without requiring additional investigation, and 40% of the proposed follow-up actions were assessed as actionable. This observation reinforces the role of our PIIS not as a fully automated solution but rather as computational support for human-on-the-loop BPI. Notably, several of the positively rated ideas aligned with the initiatives that were already underway in the organization, further highlighting the relevance of the Process Improvement Copilot.

In conclusion, RA 3 explored the research question of how to design and develop a RAG-enhanced LLM-based PIIS that supports the generation of process improvement ideas in BPI initiatives. In the spirit of the DSR paradigm, we identified the research gap, formulated five design objectives, and designed a system architecture. The RAG-enhanced architecture compensates for traditional weaknesses of LLM-based systems, such as hallucinations and lack of control over the origin of the output. We implemented the Process Improvement Copilot as a Python prototype and conducted a series of demonstration and evaluation activities: a competing artifact analysis against eight competing artifacts demonstrating competitive advantages in leveraging accumulated BPI knowledge and handling BPI initiatives with multiple process inefficiencies; 13 interviews with 16 experts confirming the importance of the problem and reduction of time and expertise in BPI initiatives; a survey validating the design objectives; and a process improvement workshop at a multinational technology conglomerate. The contributions of RA 3 are manifold. First, we propose a novel approach to automated BPI by developing a RAG-enhanced LLM-based PIIS that enables the automated generation of process improvement ideas for identified process inefficiencies. Second, we demonstrate the practical value of the Process Improvement Copilot in the workshop with a multinational technology conglomerate. Third, we encourage further exploration of the opportunities to leverage GenAI in PIISs by openly sharing our code on GitHub.

3.2 LLM-Based Redesign and Simulation of Process Models

While Section 3.1 addresses the gap of proposing context-specific, justified process improvement ideas, the next hurdle is to turn such ideas (or independently diagnosed weaknesses) into improved to-be process models. Besides being superior (i. e., better performing) to the as-is process model, these models need to be both sound (i. e., syntactically correct, as elaborated by van Dongen et al. 2006) and feasible (i. e., semantically correct). The arrangement of activities in a process model is typically referred to as the control flow, which hence represents the backbone of every process model (van der Aalst et al., 2012).

The control flow is relevant to BPI as it provides a starting point for BPI initiatives in the form of an as-is process model (Malinova et al., 2022), which is needed for incremental improvement (Davenport, 1993). To improve process models through changing the underlying control flow, multiple approaches exist, ranging from the application of business process redesign patterns (Fehrer et al., 2022; Niedermann and Schwarz, 2011), over evolutionary algorithms (Afflerbach et al., 2017), to machine learning based approaches, as demonstrated by Beheshti et al. (2023) and van Dun et al. (2023). Recently, also LLMs are utilized to automate process model generation and refinement (Kourani et al., 2024). However, extant approaches focus on automating incremental steps of BPI instead of searching for a comprehensive solution. Therefore, these approaches fall short in at least one essential BPI capability, such as process evaluation (Malinova et al., 2022), inclusion of process context factors (Moder et al., 2025), inclusion of (multiple) improvement objectives (Vergidis et al., 2006), or creativity (Gross et al., 2021).

Synthesizing computational creativity (Zhao et al., 2025) with strong reasoning capabilities (Kojima et al., 2022) and hence being able to search for BPI ideas beyond restrictive boundaries, makes LLMs a prime candidate for automating BPI. LLMs have already been employed in several BPM activities, such as explaining process models (Fahland et al., 2025), supporting the creation of process models (Ziche and Apruzzese, 2024) or even modeling processes by themselves from textual descriptions (Kourani et al., 2024; Köpke and Safan, 2025). Addressing the challenges of BPI regarding creativity and resources, and the deficiencies of extant approaches, RA 4 again follows the DSR paradigm to design, instantiate, and evaluate ABuPrOpt, a PIIS that enables the targeted generation and quantitative evaluation of improved, sound, and feasible process models. The evaluation of ABuPrOpt comprises three evaluation episodes based on Framework for Evaluation in Design Science (FEDS) (Venable et al., 2016), including a literature review, a competing artifact analysis, and a demonstration of ABuPrOpt's functionality on process models derived from five publicly available data sets. The latter evaluation episode reveals that ABuPrOpt is able to develop and quantitatively simulate improved process models in a targeted manner while considering inferred dependencies between activities.

ABuPrOpt considers three design objectives. First, it generates improved, sound, and feasible business process models. This implies a measurable improvement over the as-is model, structural soundness to avoid deadlocks or livelocks (van der Aalst, 2016; van Dongen et al., 2006), and

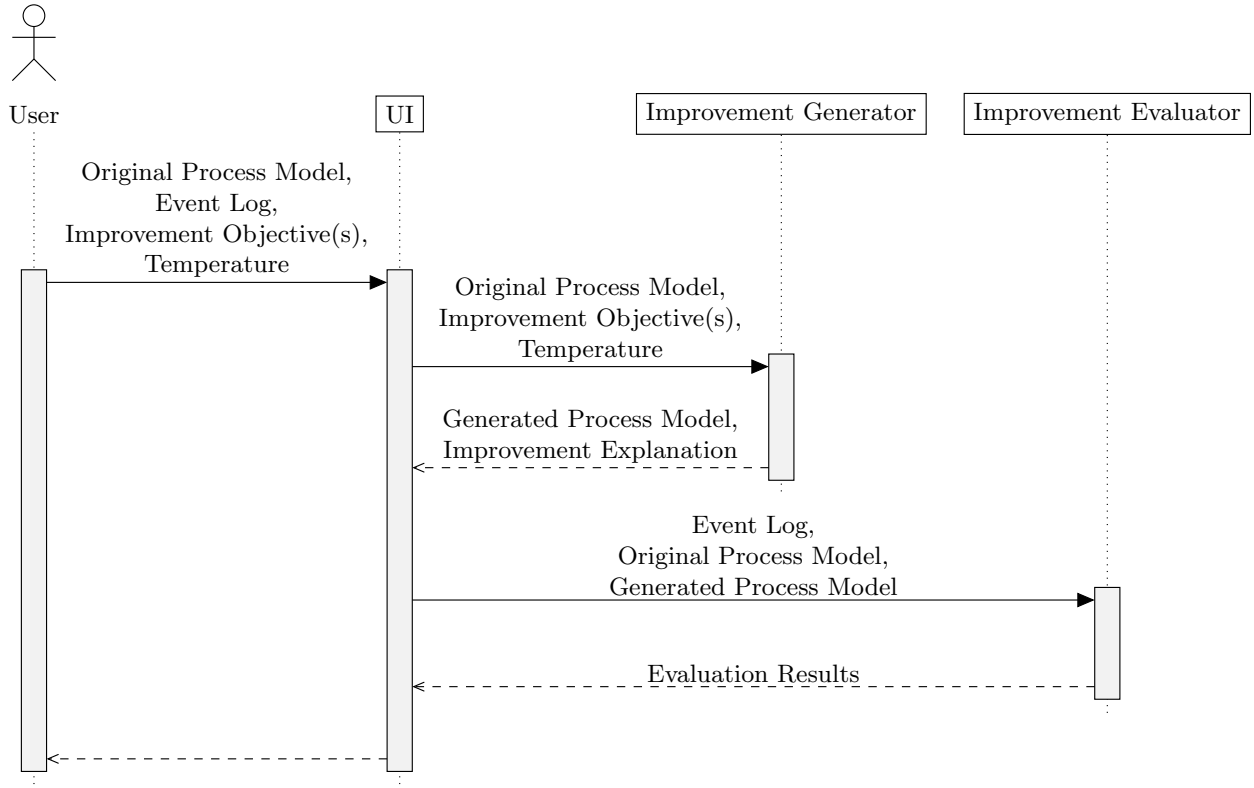


Figure 15: Sequence diagram of ABuPrOpt

semantic feasibility to respect real-world dependencies between activities (Fellmann et al., 2011). Second, it considers the desired improvement objective by featuring standard and custom objectives. In line with the devil's quadrangle (Dumas et al., 2018), ABuPrOpt can optimize for time, cost, quality, and flexibility, yet it also allows users to specify context-specific targets, such as customer-centricity (Kreuzer et al., 2020) or sustainability (Hoesch-Klohe et al., 2010). Third, it evaluates the improved business process models quantitatively to enable comparability between the original and the redesigned models.

The final design of ABuPrOpt is visualized in the sequence diagram in Figure 15. Through ABuPrOpt's User Interface (UI), the user uploads a Business Process Modeling and Notation (BPMN) 2.0 file of the original process model and a corresponding event log, specifies the improvement objective(s), and selects the LLM's temperature, which is the level of computational creativity, having an influence on the LLM's output variety (Peeperkorn et al., 2024). The original process model, the improvement objective(s), and the temperature parameter are forwarded to the improvement generator, which improves the process model and passes the improved BPMN diagram as well as the improvement explanation back to the user. The UI then passes the event log, the original process model, and the generated process model to the improvement evaluator, which assesses the time and cost performance of both the original and the generated process model. Finally, the evaluation results are presented in the UI.

The improvement generator automatically implements the specified improvement objective(s) by producing an improved, sound, and feasible process model with the help of an LLM. We selected OpenAI's *gpt-4o-2024-08-06* LLM, as the GPT-4 family serves as the underlying model family in two recent LLM-based business process modeling approaches (Kourani et al., 2024; Köpke and Safan, 2025). A key design decision concerns the representation of the original process models and the generated process models. Directly modeling in complex notations such as BPMN 2.0 is error-prone because the LLM must master both the process semantics and the formal syntax. Therefore, we convert the uploaded BPMN 2.0 process model into Partially Ordered Workflow Language (POWL) code using PM4Py (Berti et al., 2023). POWL guarantees soundness by design through partial orders and hierarchical modeling, and extends partial order graphs with control-flow operators for loops and choice (Kourani and van Zelst, 2023). However, when generating POWL process models using LLMs, it is not guaranteed that the LLM applies the POWL syntax correctly. Due to the hierarchical nature of POWL, the LLM would have to generate a highly nested POWL string. Kourani et al. (2024) resolve this complexity by developing an approach that capitalizes on the coding capabilities of LLMs. Specific Python functions to build a POWL process model are defined and passed to the LLM, together with an instruction to return code that builds a respective process model on execution. This approach ensures the safe generation of process models. Following this insight, likewise to Kourani et al. (2024), we provide Python builder functions and request executable code that constructs the generated process model. The prompt sent to the LLM, therefore, consists of three parts. First, a task specification that instructs the LLM to improve the given process model according to the chosen objective(s), second, a concise explanation of how to produce valid POWL code, adapted from Kourani et al. (2024), and third, the original process model's POWL code. The improvement generator returns the generated process model in POWL (and a corresponding explanation), which is then converted to BPMN 2.0 and forwarded to the improvement evaluator.

The improvement evaluator reveals the factual value of the suggested improvements by quantifying their impact on the original process model's performance and is split into two tightly coupled modules. The first module is a ground-truth calculator that derives the original model's time and cost from the provided event log, whereas the second module, a simulation-based approximator, estimates the same metrics for both the original and the newly generated model. Focusing on the devil's quadrangle dimensions time, cost, flexibility, and quality (Dumas et al., 2018) that internalize mutual trade-offs, we evaluate time, interpreted as cycle time per case (Dumas et al., 2018; Jansen-Vullers et al., 2008b), and operational execution cost (Jansen-Vullers et al., 2008a), while omitting quality and flexibility because they materialize outside the control-flow perspective and are hardly measurable (Dumas et al., 2018). For the ground truth, we align the event log to the original model using PM4Py (Berti et al., 2023), extract conforming cases, compute their durations, and aggregate the mean time and cost via weighted averages over variant frequencies. To compare against redesigned models that may introduce new activities or previously unseen variants, we simulate process executions, requiring the approximation of activity durations and branching probabilities. First, we derive a duration matrix of all directly-follows pairs of activities and a

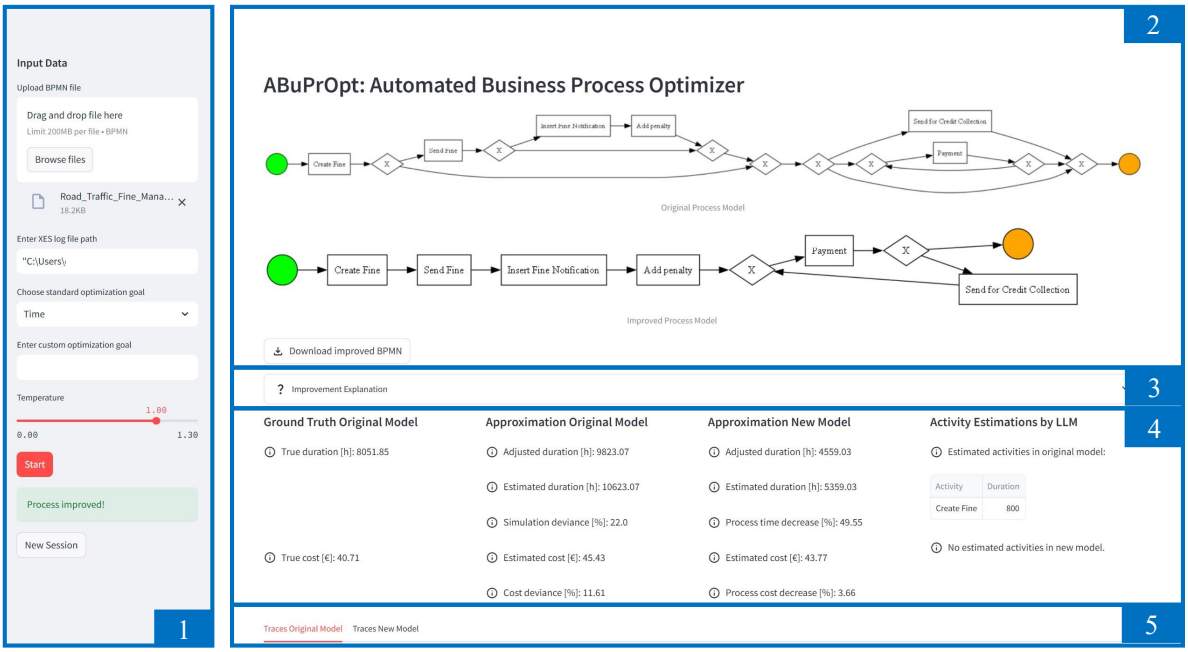


Figure 16: User interface with (1) input window, (2) visualization, (3) explanation, (4) evaluation, and (5) trace distributions

corresponding frequency matrix, and filter out weak or parallel relations. For each target activity, we compute a weighted average over significant incoming durations, thereby decoupling the target duration from predecessor-induced move/wait times and preserving only standardized move/queue plus setup/service components (Jansen-Vullers et al., 2008a). If an activity is always the first activity in the event log and hence never appears as a target, or an activity is newly introduced by the LLM, we estimate its duration and cost via an LLM. Second, for exclusive gateways, we calculate local path probabilities conditioned on the actually observed predecessor. We trim traces at the predecessor, scan suffixes for the first indirect successor (accounting for structural changes and loops), accumulate frequencies per successor and encounter, and then map successor probabilities to outgoing arcs (summing where necessary), repeating the procedure recursively when loop behavior is detected. With activity durations and gateway probabilities in place, the simulator executes 10,000 instances per evaluation: it traverses the model, samples branches at exclusive gateways based on the computed probabilities (tracking encounter counts), synchronizes parallel branches, and accumulates activity durations (and execution cost) until the end event is reached, yielding a distribution of traces and an average cycle time and execution cost.

The ABuPrOpt UI, implemented in Streamlit, guides the user through the entire workflow on a single screen and is shown in Figure 16. On the left, the user uploads the BPMN 2.0 process model to be improved, specifies the path of the corresponding XES event log, selects one or two improvement objectives (either by choosing from a drop-down list of standard objectives or typing a custom objective), and sets the LLM temperature to control the level of computational creativity

Process Model	#1 [%]	#2 [%]	#3 [%]	#4 [%]	#5 [%]
Purchase Order Handling	2.36	2.39	16.11	-20.04	-3.63
Road Traffic Fine Management	-31.76	27.59	48.30	-81.24	26.29
Hospital Billing	-4.57	27.90	-2.09	34.97	-1.53
Loan Application	-91.15	-85.82	30.70	-85.02	-85.02
Sepsis Cases	0.42	5.23	-0.60	0.34	13.52

Table 1: Simulation deviance regarding the duration of the generated process models compared to the original process models (negative percentages denote a reduction of the duration)

of the improvement generator's underlying LLM. Once these fields are confirmed, the application reads the event log, invokes the improvement generator, and then passes both the original and the redesigned model to the improvement evaluator. The UI finally presents both models below each other, provides the LLM's textual explanation of the changes, and reports key indicators: average cycle times and execution costs, trace-frequency distributions, durations per trace, the deviation of the simulation of the original model, and the net reduction in cycle time and execution cost achieved by the redesign.

To demonstrate the functionality of ABuPrOpt, we improved five process models derived from five publicly available real-world XES event logs: the Purchase Order Handling event log (van Dongen, 2019), the Road Traffic Fine Management event log (de Leoni and Mannhardt, 2015), the Hospital Billing event log (Mannhardt, 2017), the Loan Application event log (van Dongen, 2017), and the Sepsis Cases event log (Mannhardt, 2016). Although these five event logs contain temporal data at the event level, none of the event logs contain cost attributes at the event level. Therefore, we instructed *gpt-4o-2024-08-06* to generate a mapping based on the Road Traffic Fine Management event log that contains estimations of process execution costs on the activity level. In this way, we enhance the event log with artificial cost information, which we utilize to evaluate the cost dimension. Furthermore, we generate a BPMN 2.0 conform process model from each event log using PM4Py's inductive miner based on the five most frequent variants in the respective event log. The resulting process models feature the most common BPMN 2.0 elements, such as activities, exclusive gateways, parallel gateways, and sequence flows.

Five iterations of reducing the cycle time of the five process models (with the default temperature of 1) resulted in 25 measurements, listed in Table 1. Out of 25 improvement cycles, 12 resulted in an actual decrease in the duration. The proportion of generated models with a decreased duration took varying magnitudes, depending on the original process model. For example, four out of five cycles generated from the Loan Application process model yielded a reduced duration, compared to only one out of five cycles towards the Sepsis Cases process model.

The improvement cycles that did not reduce the duration of the given process model and hence did not fulfill the given improvement objective can be traced back to eliminating optional trajectories by removing skips of activities. Moreover, there were also some generated models where the improvement generator rearranged activities and gateways in one place in a helpful way, while at

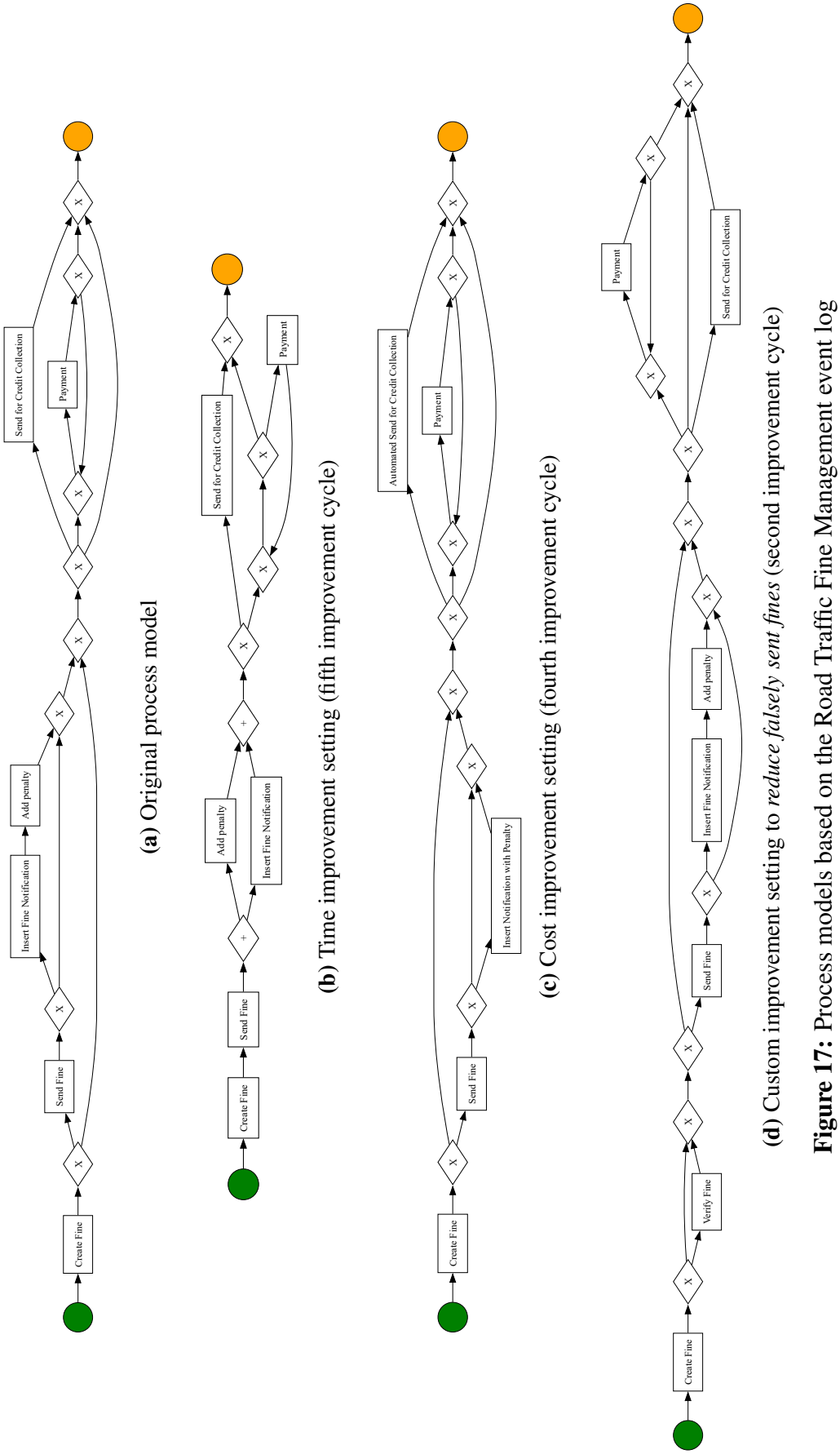
Process Model	#1 [%]	#2 [%]	#3 [%]	#4 [%]	#5 [%]
Road Traffic Fine Management	19.62	8.53	37.68	-39.00	31.58

Table 2: Estimated cost performance of generated process models in comparison to the estimated cost performance of the original process model

the same time, it changed other process parts in an unhelpful way, and hence, no net reduction in the cycle time of the process model could be achieved. For example, in the fifth improvement cycle of the Road Traffic Fine Management process (see Figure 17a for the original model and Figure 17b for the generated model), the improvement generator suggests parallelizing the activities "Insert Fine Notification" and "Add Penalty" in the front portion of the model. Additionally, it made the self-loop around the "Payment" activity optional. However, it eliminated all other optional paths from the process model, referring to them as redundant and arguing that removing them could facilitate more rapid decision-making. Adversely, this measure forces every process instance to run through all activities, even though the original process model indicated that some of the activities can be legitimately skipped. This misjudgment of the improvement generator was observed in most cases where the cycle time of the respective process model increased.

Table 2 shows the result of five improvement cycles on the Road Traffic Fine Management process targeted at reducing the process execution cost. The fourth improvement cycle leads to a decrease in process execution cost by 39.00%. The reduction in execution cost in this process model (Figure 17c) was achieved by two measures, the first connecting the activities "Insert Fine Notification" and "Add Penalty" into a single step. The improvement generator justifies this approach with the rationale that synthesizing those steps can reduce overhead costs caused by isolated execution. Additionally, the improvement generator decided to automate the activity "Send for Credit Collection", arguing that this measure diminishes the necessity of human intervention. Additionally, it states that the automated activity can be triggered based on conditions, which minimizes costs caused by manual monitoring. In the improvement cycles where execution costs increased, the reasons were similar to those already observed for the improvement objective time. In most cases, the cause of the poorer cost performance was the removal of optional paths, sometimes in parallel with beneficial measures, such as parallelization, consolidation, or automation.

To test ABuPrOpt with a custom improvement objective, we prompted it five times to *reduce falsely sent fines* in the Road Traffic Fine Management process. The most frequently implemented improvement comprised adding a verification step directly after creating a new fine to enable early detection of incorrect fines (Figure 17d). Another idea was an approval step by a manager before the fine is sent out. Alternatively, the optionality of the "Send Fine" and "Insert Fine Notification" activities was removed, as skipping these could lead to errors in the fine according to the improvement generator. In all improvement cycles, the improvement generator detected suitable locations for improvement prior to the activity "Send Fine" and implemented measures that specifically targeted the objective.



(a) Original process model

(b) Time improvement setting (fifth improvement cycle)

(c) Cost improvement setting (fourth improvement cycle)

(d) Custom improvement setting to *reduce falsely sent fines* (second improvement cycle)

Figure 17: Process models based on the Road Traffic Fine Management event log

In summary, RA 4 showed that LLMs and a simulation based on event logs can elevate BPI from isolated control-flow tweaks to a holistic, data-driven redesign cycle that yields improved, sound and feasible process models. Following the DSR paradigm we derived three design objectives and realized them in a Python prototype that converts an uploaded BPMN 2.0 model into POWL, prompts *gpt-4o-2024-08-06* to return executable builder functions for the redesigned model, and then simulates both the original and generated process model with activity-level durations and gateway probabilities inferred from the provided event log. The evaluation based on FEDS comprised a literature review, a competing artifact analysis against five related approaches and a demonstration on five publicly available logs with different improvement objectives, confirming that ABuPrOpt delivers measurable gains while maintaining syntactic and semantic correctness. The study contributes to BPI research by introducing a PIIS with a solution space expanding beyond potential boundaries by process input data or externally defined instructional frameworks, by incorporating both standard and custom improvement objectives, and by coupling the generation of redesigned process models with a simulation that quantitatively benchmarks these models on time and cost. For practitioners, we provide a prototype that helps organizations find and compare suitable improvement ideas for their processes and enables the judgment of each redesign's business case on multiple performance figures rather than intuition alone.

4 AI-Supported Process Monitoring

Sections 2 and 3 demonstrated how GenAI can improve business processes by revealing process execution reality from unstructured data and by suggesting and evaluating redesigned process models. However, an equally powerful lever of BPI is the automation of operational work itself. Delegating repetitive activities to AI agents rather than human employees enables organizations to deliver services around the clock, at marginal cost close to zero, and with a level of consistency that manual execution rarely achieves (Adamopoulou and Moussiades, 2020). For example, customer service chatbots scale to millions of simultaneous inquiries without decreasing customer satisfaction since they conveniently provide assistance and access to information, independent of the availability of human agents (Brandtzaeg and Følstad, 2017). Another example is the automation of recruiting processes, where linguistic analysis promises to uncover applicants' behavioral traits more quickly and more truthfully than traditional questionnaires (Boyd and Pennebaker, 2017; Newman et al., 2003; Stachl et al., 2020). Automation, therefore, complements process discovery, analysis, and redesign by unlocking new service levels and reducing operational cost, provided that the automated behavior follows the intended organizational procedures and ethical standards. Ensuring such alignment is not trivial. Chatbots leaking customers' personal data due to improper authentication can destroy customer trust despite flawless natural language generation (Gunson et al., 2011). Likewise, misclassifying applicants' cooperativeness because they pretend having certain personality traits jeopardizes fair hiring decisions (Birkeland et al., 2006; Rosse et al., 1998). Consequently, organizations require monitoring instruments to examine AI-driven activities and assess both compliance and performance of the underlying workflow.

This section addresses those challenges with two studies situated in the process monitoring phase of the BPM lifecycle. Section 4.1 (RA 5) develops a chatbot evaluation workflow that quantifies a chatbot's ability to adhere to organizations' business processes. Building upon more than 500,000 Twitter conversations from three companies, we convert these dialogues into XES event logs using the NLI-based extraction pipeline presented in Section 2.1. We then train a customer service chatbot for each company, and align each chatbot's dialogue traces on unseen data with traces derived from the training data using standardized conformance checking metrics in process mining. The resulting conformance scores reveal each chatbot's overall process compliance and thus enable an objective benchmarking of different model versions before deploying chatbots at scale.

Section 4.2 (RA 6) tackles a complementary challenge in recruitment by investigating whether cover letters, if analyzed by an appropriately trained classifier, predict true cooperativeness more robustly than psychometric tests when applicants have an incentive to fake being cooperative. In an online experiment with 400 participants, we elicit the participants' true cooperativeness using a public goods game. Under baseline and incentive conditions, we collect 3,000-character long self-descriptions (in the spirit of a cover letter) and 10-item Big Five personality questionnaire responses, and train both linguistic and psychometric classifiers. Our findings show that text-based models significantly outperform questionnaire-based models once salient incentives to fake arise and can detect the presence of such incentives.

4.1 Quantifying Chatbots' Adherence to Business Processes

Proactive customer orientation is a proven driver of customer value (Blocker et al., 2011). As customers increasingly expect to choose when and how they interact with a company, organizations need to deliver omni-channel experiences (Hosseini et al., 2018). Digital communication channels, such as email, social media, and instant messaging, have therefore become indispensable, and the ever-increasing volume of requests through these channels requires organizations to scale service operations beyond what human agents alone can deliver (Følstad and Brandtzæg, 2017).

Chatbots offer a feasible, affordable, and scalable solution since they can handle large fractions of repetitive inquiries while maintaining the customer orientation that is essential to uphold purchase intentions (Poddar et al., 2009). Chatbots are already employed by public agencies (Androutsopoulou et al., 2019) and by companies such as Amazon (Følstad and Brandtzæg, 2017), Deutsche Telekom (Simon, 2019), and Lufthansa (Ukpabi et al., 2019). According to Brandtzaeg and Følstad (2017), the primary motivation for users to engage with chatbots is the immediate access to assistance they provide, regardless of staff availability.

However, a linguistically fluent chatbot alone does not guarantee business value. A customer-facing chatbot must comply with organizational or regulatory requirements, such as authenticating a customer before disclosing sensitive data, following escalation paths when a request exceeds the chatbot's competence, and collecting all required information before handing a conversation to a human agent. Figure 18 contrasts two dialogues. On the left, the chatbot provides a seemingly appropriate answer, but leaks personal information because it neglected authentication. On the right, the bot recognizes an order number as a valid surrogate for the missing customer number, gathers the necessary facts, and only then releases the details. This example shows that organizations cannot assess chatbots solely on conversational quality alone, but need evidence that the learned behavior is process-compliant.

Despite this requirement, existing chatbot evaluation approaches focus almost exclusively on linguistic metrics. What remains missing is a systematic way to assess whether a chatbot has internalized organizations' business processes. To close this gap, RA 5 (Kecht et al., 2023) introduces an approach that quantifies a chatbot's adherence to business processes with standardized conformance-checking metrics from process mining (van der Aalst, 2011). We develop this approach in line with the DSR methodology of (Peffers et al., 2007). Guided by four design objectives, the artifact scales to multiple chatbots and a vast number of process instances within the training data, transforms existing and replayed dialogues into a standardized event log without manual annotation, quantifies a chatbot's conformance both with processes implicitly contained in its training data and with an explicit normative process model, and supports the meaningful assessment of a single chatbot as well as the comparison of several chatbots. To evaluate the approach, we follow the framework of Sonnenberg and vom Brocke (2012a,b) and conduct a competing artifact analysis, develop a prototype in Python on top of PM4Py (Berti et al., 2023), and assess chatbots trained on real-world datasets of customer service conversations using our artifact.

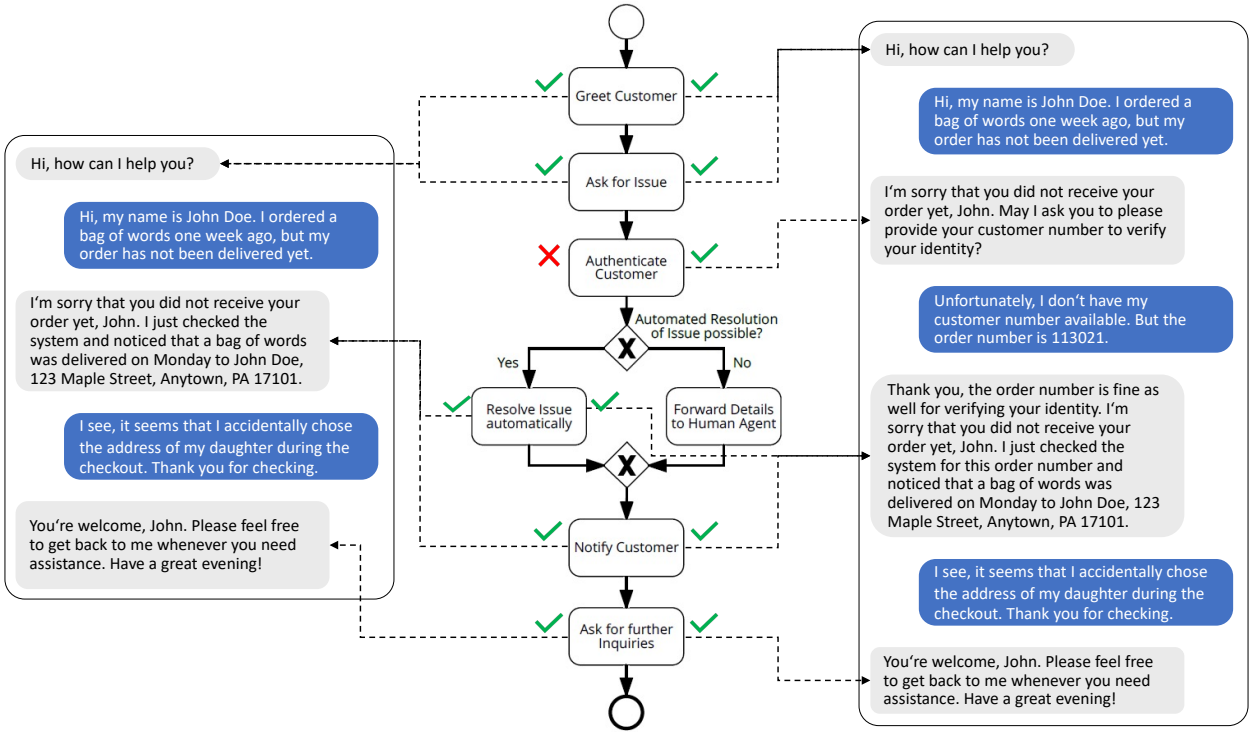


Figure 18: Exemplary conversations with (1) a chatbot giving a suitable response to the customer's inquiry while failing to achieve process conformance (left) and (2) a chatbot giving a suitable response to the customer's inquiry while achieving process conformance (right)

Figure 19 shows how our approach applies conformance checking to quantify a chatbot's ability to learn and adhere to organizations' business processes. Chatbots are trained on an existing set of conversations between humans and humans, for example, customers and customer service agents. Apart from noisy process instances, these conversations adhere to a normative process model. If no such model exists, it can be derived from event logs synthesized from the conversations, as Kecht et al. (2021) show. By reporting the trace alignment (Rogge-Solti et al., 2016) between an event log constructed from conversations the chatbot has not seen during the training process and an event log from the same conversations in which we replaced the customer service agents' responses with chatbot-generated responses, the approach can assess the chatbot's overall ability to learn and adhere to business processes.

To enable a breakdown on a particular process variant and the comparison against normative process models, we calculate four metrics (ranging between 0 and 1) that can be interpreted in practice as follows. *Fitness* (Rozinat and van der Aalst, 2008) measures to which extent the chatbot does not introduce new process variants. *Precision* (Muñoz-Gama and Carmona, 2010) describes whether the process model disallows the creation of new process variants by the chatbot. *Generalization* (Buijs et al., 2012) ensures that the process model does not overfit by capturing each trace of the event log as a separate path. *Simplicity* (Buijs et al., 2012) denotes whether the process model has low complexity.

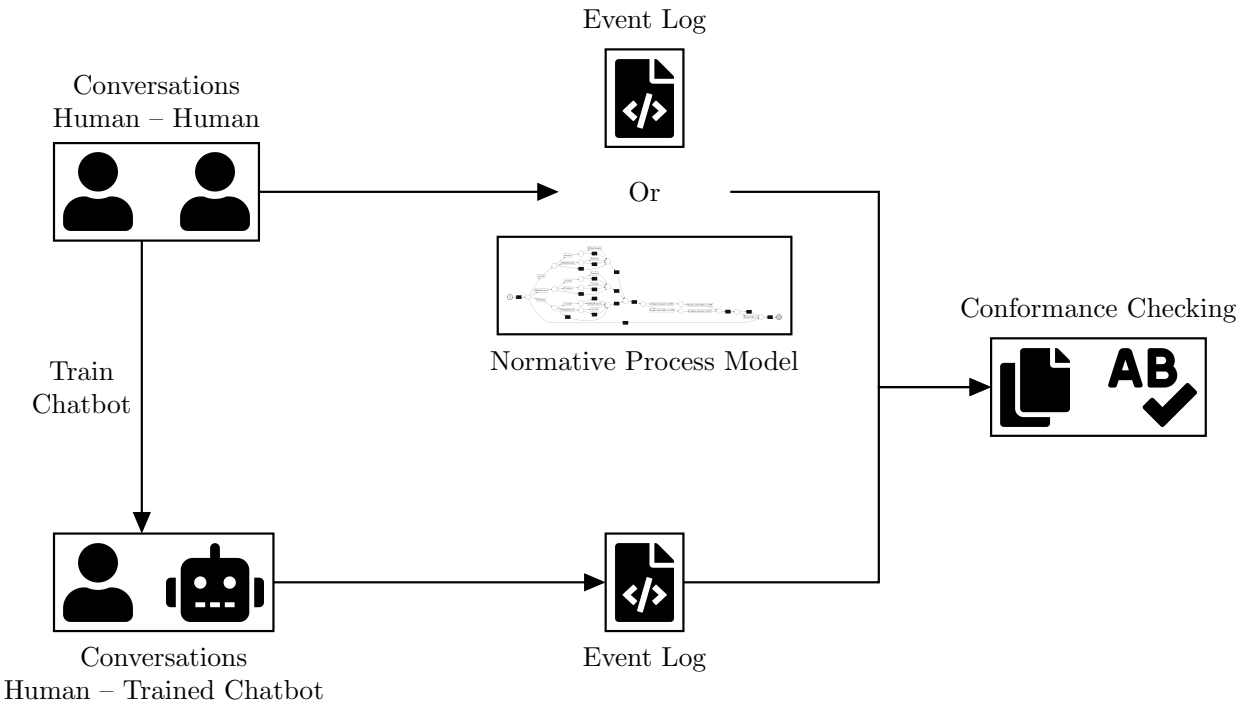


Figure 19: Overview of the approach for quantifying chatbots' adherence to business processes

Our approach complements the classical supervised machine learning evaluation workflow with a business process perspective. To enable an unbiased evaluation, the classical supervised machine learning evaluation workflow evaluates a model's performance on data it has not seen during the training process, typically referred to as test data. Figure 20 visualizes our proposed workflow that can be described as follows:

1. Train the chatbot on the training data.
2. Convert the training data to an XES event log using, for example, the approach from Kecht et al. (2021).
3. Replay the conversations in the training data by replacing the agents' responses with a chatbot-generated response and convert the resulting dataset to an XES event log using, for example, the approach from Kecht et al. (2021).
4. (Optional) If a quantification of the chatbot's ability to adhere to a normative process model is desired, specify the normative process model. However, in case there is no normative process model yet, a discovered process model using the discovery algorithms of "PM4Py", e. g., the Alpha Miner (van der Aalst et al., 2004), the Inductive Miner (Leemans et al., 2013), or the Heuristics Miner (Weijters et al., 2006), can either serve as a suitable proxy or support the specification of the normative process model.
5. Depending on whether a quantification of the ability to learn business processes from the training data or to adhere to a normative process model is desired, compare the event logs resulting from Step 3 either to the proxy models discovered in Step 4 or to the specified normative process

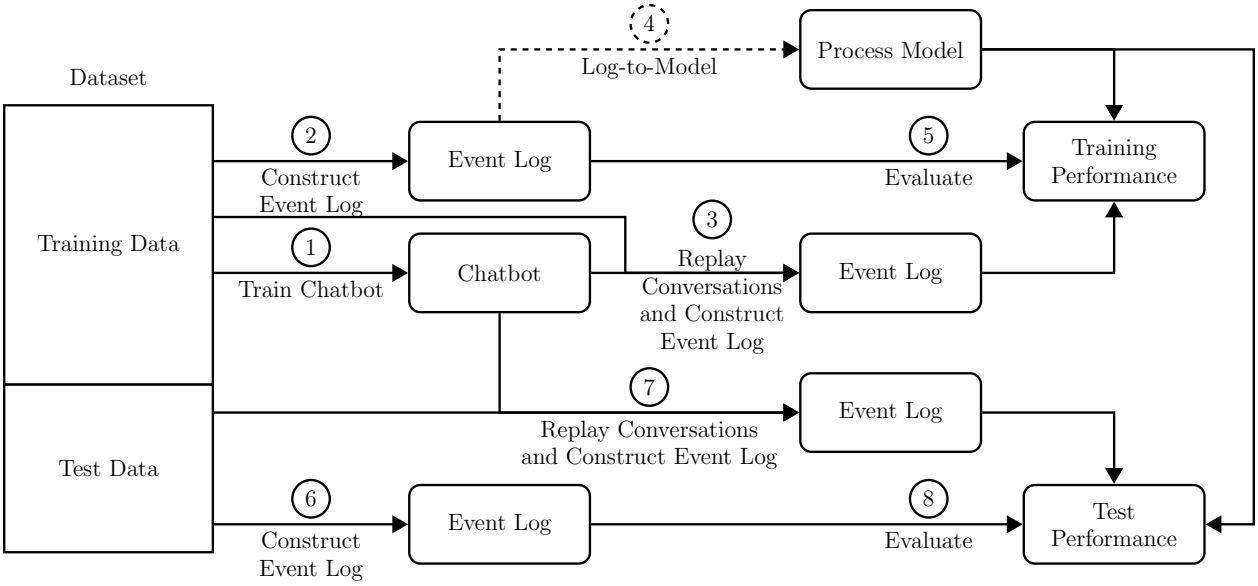


Figure 20: Evaluation workflow for quantifying chatbots' adherence to business processes

model using the four metrics fitness, precision, generalization, and simplicity. Additionally, we propose to measure the similarity of the event logs resulting from Steps 2 and 3 using trace alignment.

6. Convert the test data to an XES event log, e. g., using the approach from Kecht et al. (2021).
7. Replay the conversations in the test data by replacing the agents' responses with a chatbot-generated response and convert the resulting dataset to an XES event log using, for example, the approach from Kecht et al. (2021).
8. Similar to Step 5, depending on whether a quantification of the ability to learn business processes from the training data or to adhere to a normative process model is desired, compare the event logs resulting from Step 7 either to the proxy models discovered in Step 4 or to the specified normative process model and apply trace alignment on the event logs resulting from Steps 6 and 7.

To demonstrate the applicability and usefulness of our artifact in practice, we show how our approach enables investigating practical questions by reporting the corresponding process mining metrics. We chose the Twitter dataset Thought Vector (2017) using the Tweets and replies to and from AmazonHelp, AppleSupport, and SpotifyCares for a comprehensive evaluation. We preprocessed the dataset with the same steps as in the event log construction approach described in Kecht et al. (2021), split the data of each company into a training dataset (55% of the data) and a test dataset (45% of the data), and trained a chatbot for each company. We then used these chatbots to replace the customer service agents' responses with a chatbot-generated response in the training and test datasets, resulting in four conversation files for each company, i. e., training conversations, test conversations, replayed training conversations, and replayed test conversations. Finally, we converted each file to a corresponding XES event log using the approach from Kecht et al. (2021).

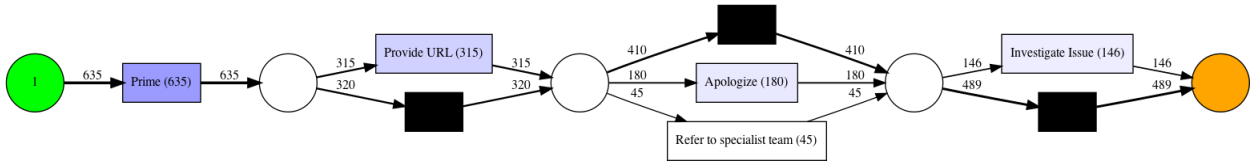


Figure 21: Process map of AmazonHelp discovered by the Inductive Miner - infrequent after selecting the ten most frequent process variants for inquiries regarding Amazon Prime

Event Log	Replay Fitness	Precision	Generalization	Simplicity
Training Event Log	1.000	1.000	0.931	0.684
Replayed Training Event Log	0.944	0.912	0.960	0.684
Test Event Log	1.000	1.000	0.921	0.684
Replayed Test Event Log	0.941	0.904	0.956	0.684

Table 3: Log-to-model evaluation metrics for the discovered process model in Figure 21

To investigate how the ability of the chatbot of AmazonHelp to deal with inquiries regarding Amazon Prime can be quantified, we first eliminate noisy process instances. To this end, we exemplary assume that the ten most frequent variants in the training event log suitably represent the desired process variants. Applying the Inductive Miner - infrequent (Leemans et al., 2014) yields the process model depicted in Figure 21. Table 3 lists the replay fitness, precision, generalization, and simplicity for the four event logs.

Since the model achieves a replay fitness and precision of one, all traces in the original logs (the training and test event logs contain the same variants) can be replayed. No unseen variants are possible in the process model. The generalization of 0.931 indicates that the process model does not overfit to the training event log. Furthermore, the replay fitness of 0.944 and 0.941 on the replayed event logs represents to which extent the chatbot did not introduce new process variants while dealing with inquiries regarding Amazon Prime. The precision values of 0.912 and 0.904 and the generalization values of 0.960 and 0.956 (that are higher than the values of the corresponding original logs) show that the chatbot did not execute all variants in the model. With the same variant filter, the trace alignment between the training and the replayed training event log equals 0.720. In contrast, the alignment between the test event log and the replayed test event log equals 0.724. Since there is no significant difference between the values on the replayed test event log and the replayed training event logs, the chatbot's underlying model seems to generalize well for unseen process instances and, thus, does not overfit to the training data. From these results, we conclude that the chatbot for AmazonHelp profoundly learned how to handle inquiries regarding Amazon Prime.

In conclusion, RA 5 presents an approach that quantifies chatbots' ability to learn and adhere to organizations' business processes. Following the DSR methodology of Peffers et al. (2007), we derived four design objectives, implemented the solution in Python on top of PM4Py, and automated

every step from event log construction to conformance calculation. The evaluation was conducted in line with the framework of Sonnenberg and vom Brocke (2012a,b). We compared our solution with three competing artifacts, instantiated a prototype, and assessed its applicability to real-world data by training three chatbots on more than 500,000 Twitter conversations from AmazonHelp, AppleSupport, and SpotifyCares. The experiments show how the approach quantifies a chatbot's overall ability to learn business processes from the training data, measures its adherence to a particular process variant, and contrasts its behavior with a normative process model. The contribution is threefold. First, we bridge the gap between NLP research and process science by demonstrating how process discovery and conformance checking extend classical chatbot evaluation. Second, we validate the approach with real customer-service data, yielding concrete insights into the process compliance of three chatbots. Third, we release the full implementation on GitHub, enabling scholars and practitioners to reuse and extend the work. By mitigating the current challenges practitioners face when deploying chatbots, such as ensuring compliance with organizational or regulatory requirements, our approach supports the application of chatbots within customer-centric BPI initiatives.

4.2 NLP-Based Personality Assessment of Job Applicants

Proactive, data-driven process monitoring does not end with checking whether chatbots follow organizations' business processes, but also extends to the people operating and improving those processes. Many organizations now delegate early stages of recruiting to AI services that screen resumes, parse cover letters, or even interact with applicants before a human enters the loop. However, selecting the best available candidate for a vacancy is crucial for organizational success and extends far beyond matching resumes and cover letters to job requirements. Organizations have long been trying to capture applicants' personality (Scepura, 2020; Varela et al., 2004), for example, their disposition for being a good team player, which has become an increasingly sought-after trait in many industries (Chen and Gong, 2018; Lazear and Shaw, 2007). To gauge a candidate's personality, organizations frequently use self-reported psychometric tests like the 'Big Five' (Goldberg, 1990), which attempt to distill people's personality traits from their answers to a number of Likert-scale questions. For instance, the Big Five trait of *Agreeableness* has recurrently been found to be a good predictor of a person's disposition for being a cooperative team player (Kagel and McGee, 2014; Koole et al., 2016; Volk et al., 2011). Yet, psychometric tests like the Big Five have a critical flaw: they are not robust to the presence of incentives to *pretend* having certain personality traits (Morgeson et al., 2007; Tett and Simonet, 2021). Put simply, an applicant who anticipates that *looking* like a good team player increases her chances of being hired, will make sure to score high on *Agreeableness*. More often than not, job applicants will have a good idea of what recruiters expect, and thus an incentive to sugarcoat their personality.

Recent advances in AI, and in particular, NLP, such as Meta's BART model (Lewis et al., 2020a) and OpenAI's GPT-3 (Brown et al., 2020), have opened an enticing new path for the prediction of personality (Boyd and Pennebaker, 2017; Stachl et al., 2020). Most importantly, language – both

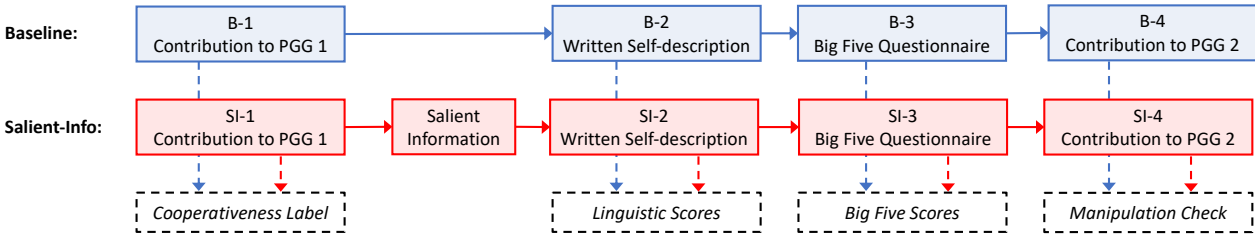


Figure 22: Design of the online experiment to measure subjects' cooperativeness in the absence and presence of salient incentives to fake being cooperative

spoken and written – has the potential to be considerably more robust than psychometric measures against people's temptation to fake their personality (Newman et al., 2003). Even if job applicants – in an attempt to please the recruiter – managed to modify *what* they say in a cover letter, extant research suggests that they may find it substantially more difficult to modify *how* they say it (Bond and Lee, 2005; Hancock et al., 2007; Hauch et al., 2015; Newman et al., 2003; Zhou et al., 2004).

Hence, RA 6 (Kecht et al., 2022) studies the question whether cover letters – analyzed by an appropriately trained AI – are better suited to assess applicants' personality traits than easy-to-fake psychometric tests. To answer this question, we conducted a controlled online experiment with 400 participants. The results show that, once salient incentives to fake exist, linguistic classifiers significantly outperform psychometric ones and that a fine-tuned language model can detect incentives to fake in people's self-descriptions. These findings demonstrate a viable path towards more tamper-resistant, AI-supported hiring processes and illustrate how rigorous experimental evaluation can complement process monitoring in the BPM lifecycle.

The study employs an experiment to construct a dataset with (1) individual-level data on people's self-descriptions, their answers to a Big Five questionnaire, and an incentivized measure of their actual cooperativeness, and (2) exogenous variation of people's incentives to fake their personality. As shown in Figure 22, the subjects were first randomly assigned to a *Baseline* or a *Salient-Info* group with a probability of 75% and 25%, respectively. In stage one, we divided subjects into groups of four players to elicit their *true* cooperativeness in a public goods game (Isaac and Walker, 1988). In this game, which is the most widely-used measure of cooperative behavior in experimental economics, each player decides how many of 20 points to invest into a joint project. Each player's payoff function reads: $\pi_i = 20 - g_i + 0.4 \cdot (g_i + \sum_{j=1}^{3} g_j)$, where g_i denotes the player's contribution, g_j are the other players' contributions, and 0.4 is the marginal payoff of contributing to the joint project. The socially optimal outcome $\pi_i = 32$ is achieved when all players contribute all their 20 points to the joint project. However, individually each player is tempted to unilaterally increase their individual payoff to $\pi_i = 44$ by contributing 0 points while the other players continue contributing 20 points. Hence, the setup resembles a teamwork situation in which individual team members face a dilemma between what is best for the team as a whole, and what is best for them individually. We interpret subjects' contributions g_i as our discrete measure of their true cooperativeness. Immediately afterwards, both groups were told that three additional personality tests would follow, while subjects in the Salient-Info group also learned that belonging to the 40% most

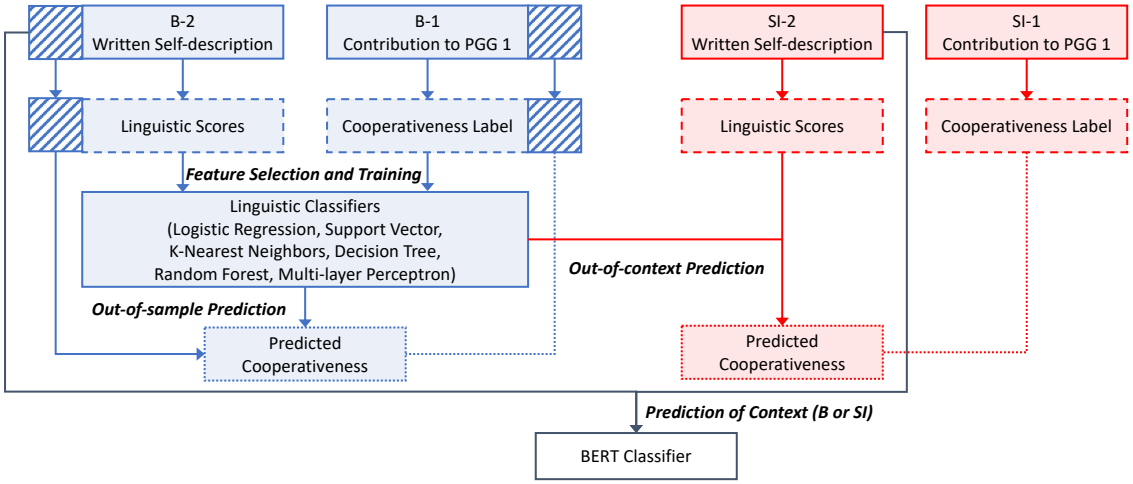


Figure 23: Machine learning approach for out-of-sample prediction, out-of-context prediction, and prediction of the context

cooperative participants (decided by a committee) would yield bonus of €10, creating a salient incentive to fake being cooperative in the subsequent stages. In stage two, subjects wrote a self-description of approximately 3,000 characters, resembling the length of a typical one-page cover letter. In stage three, subjects performed a self-reported 10-item Big Five personality test (Gosling et al., 2003), containing two items per personality trait measured on a 7-point Likert scale. In stage four, subjects played a second public goods game and made an unconditional contribution decision similar to stage one, which served as a manipulation check. After the main experimental stages, we finally elicited a conditional cooperation test (Fischbacher et al., 2001), subjects' beliefs about their position in the cooperativeness ranking, and a socio-demographics questionnaire.

We programmed the experiment in oTree (Chen et al., 2016) and ran 17 online sessions with 400 university students recruited via experimenTUM. After excluding ten subjects with non-compliant written self-descriptions and 94 non-native speakers, the final dataset comprises 296 subjects (217 Baseline, 79 Salient-Info), aged 24 years on average and predominantly at high-school or bachelor level, hence representing a group likely to soon enter the job market. Randomization and manipulation checks confirm the experiment design's validity. Contributions in the first public goods game do not differ between the groups (Mann-Whitney U test: $p = 0.695$; Baseline $\bar{\mu} = 9.06$, Salient-Info $\bar{\mu} = 8.66$). In the second game, subjects in the Salient-Info group contribute significantly more than Baseline peers (Mann-Whitney U test: $p < 0.001$; Baseline $\bar{\mu} = 9.00$, Salient-Info $\bar{\mu} = 13.27$), demonstrating that the bonus information effectively led the Salient-Info group to fake being cooperative.

We compare the predictive power of written self-descriptions and answers to the Big Five personality test as illustrated in Figure 23. For both the Baseline and Salient-Info group, we compile a dataset whose features are the 3,000-character self-descriptions collected in stage two, while the target is a binary cooperativeness label, i. e., two classes denoting whether a subject's contribution in the first public goods game is above the median of the *Baseline* group. The textual pipeline be-

gins by scoring every document with the 2015 German Linguistic Inquiry and Word Count (LIWC) dictionary, yielding 97 stylistic and semantic categories (Pennebaker et al., 2015). Since superfluous features inflate noise and degrade learning, we apply a filter method. Within the Baseline group, the LIWC categories *Sadness*, *Future focus*, and *Periods* are negatively correlated with subjects' true cooperativeness at the 10% level, whereas *3rd pers plural*, *Common Adverbs*, *Anxiety*, *Health*, *Drives*, and *Religion* show positive correlations at the 10% level. Consequently, we selected these nine categories as features for the training of our linguistic classifiers. For the Big Five scores, we find that the statements "I see myself as ..." (1) "Open to new experiences, complex", (2) "Conventional, uncreative", (3) "Dependable, self-disciplined", and (4) "Sympathetic, warm" are significantly correlated with subjects' true cooperativeness at the 10% level. Therefore, we selected these four items as numeric features for the psychometric classifiers.

As machine learning models, we used six classifiers for binary classification from the Python library "scikit-learn" (Pedregosa et al., 2011). We trained the linguistic classifiers on the nine selected LIWC categories and identified the optimal hyperparameter set for each classifier using a nested cross-validation approach to maximize the out-of-sample performance. To this end, we held out 20% of the data (visualized by the shaded rectangles) in each of the five iterations of the nested cross-validation approach and used this data to calculate the out-of-sample performance by comparing the predicted binary cooperativeness with the previously assigned binary cooperativeness labels. The training of the psychometric classifiers followed an analogous procedure. As a benchmark for our predictions, we used four standard dummy classifiers provided by "scikit-learn": (1) the Dummy Minority Classifier, which always predicts that a given player's cooperativeness is above the median, (2) the Dummy Majority classifier, which always predicts that a given player's cooperativeness is below or equal to the median, (3) the Dummy Uniform Classifier, which tosses a fair coin to decide whether a given player's cooperativeness is above the median, and (4) the Dummy Stratified Classifier, which determines each label randomly based on the training set's class distribution and therefore provides a tougher benchmark than the other dummy classifiers. Moreover, from a business perspective, the Dummy Stratified Classifier represents a recruiter who decides whether *current* applicants are above the median in terms of their cooperativeness or not, based on her knowledge about the distribution of cooperativeness among *past* applicants. We investigated the performance of the initial six classifiers – compared to the Dummy Stratified Classifier – in two distinct situations: first, in the absence of salient incentives to fake being cooperative (out-of-sample predictions on data from the *Baseline* group), and second, in the presence of salient incentives to fake being cooperative (out-of-context predictions on data from the *Salient-Info* group).

Finally, we predict the context (i. e., whether a subject had salient incentives to fake being cooperative or not) based on the raw text of the self-descriptions using a pre-trained German BERT language model. This allows us to exploit the full potential of raw text data rather than reducing it to the LIWC scores used for the out-of-sample and out-of context predictions. For the Big Five scores, there was no need to transform the answers from the questionnaire to numeric features. Therefore, there is no point in an analogous step based on the Big Five scores.

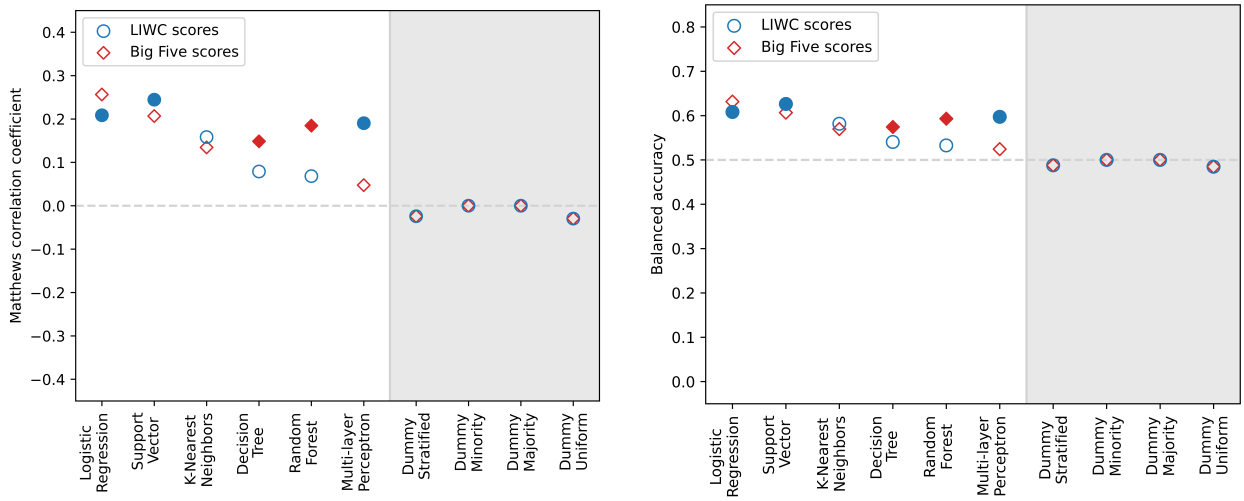


Figure 24: MCC and balanced accuracy for predictions in the absence of salient incentives to fake being cooperative (out-of-sample performance)

Figure 24 visualizes the aggregated MCC and balanced accuracy over all five iterations of the outer cross-validation loop for each type of classifier including the four dummy classifiers. The red diamonds represent the psychometric classifiers, whereas the blue circles represent the linguistic classifiers. Four psychometric classifiers and four linguistic classifiers achieve an MCC close to 0.2, indicating a weak positive relationship between subjects' predicted cooperativeness and their true cooperativeness. The balanced accuracy of these eight classifiers ranges between 0.57 and 0.63. By definition, the Dummy Minority Classifier and the Dummy Majority Classifier achieve an MCC of 0 and a balanced accuracy of 0.5, whereas the Dummy Uniform Classifier achieves an MCC of -0.03 and a balanced accuracy of 0.48. To test whether our classifiers' predictions are significantly better than the Dummy Stratified Classifier (which reaches an MCC of -0.02 and a balanced accuracy of 0.49), we conducted pairwise McNemar's tests between the classifiers' predictions on subjects' true cooperativeness based on their Big Five and LIWC scores and those of the Dummy Stratified Classifier (Dietterich, 1998; McNemar, 1947). Two psychometric classifiers and three linguistic classifiers predict significantly better than the Dummy Stratified Classifier, indicated with a filled marker in Figure 24. For the subsequent analysis in the presence of salient incentives to fake being cooperative, we selected the hyperparameter set for each classifier that achieved the highest MCC in the outer cross-validation loop and retrained each classifier with the entire data of the *Baseline* group. Thereby, we reduce the impact of the train-test split introduced by the cross-validation approach.

To study the effect of salient incentives to fake being cooperative on the performance of linguistic scores and Big Five scores, we used the scores from the *Salient-Info* group as features and predicted whether subjects' true cooperativeness is above the median or not. As Figure 25 shows, the six classifiers based on Big Five scores perform worse than the Dummy Stratified Classifier, whereby this difference is statistically significant in three cases. On the other hand, four of six classifiers based on LIWC scores achieve a higher MCC and a higher balanced accuracy than the Dummy

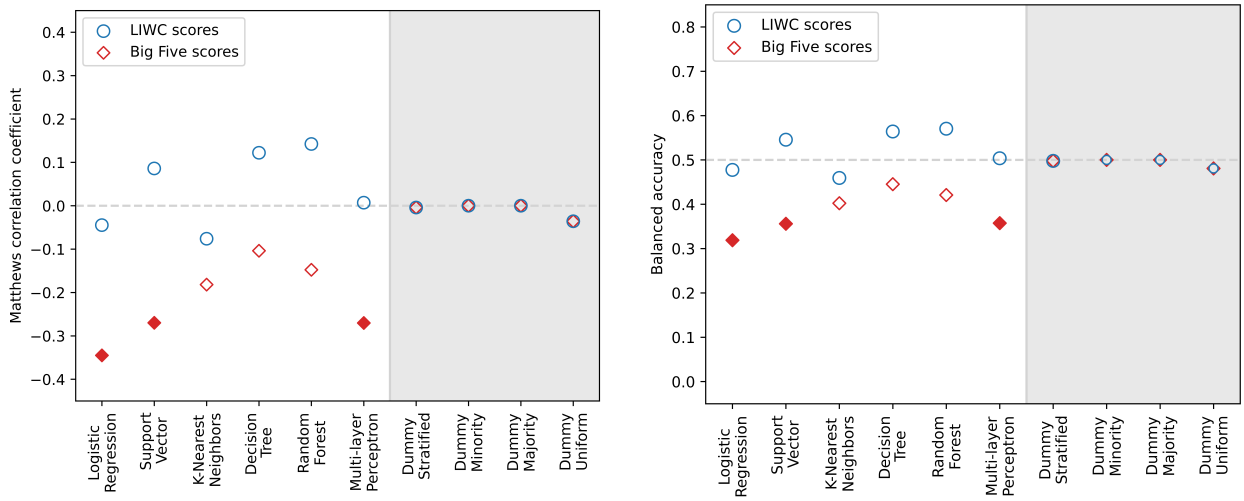


Figure 25: MCC and balanced accuracy for predictions in the presence of salient incentives to fake being cooperative (out-of-context performance)

Stratified Classifier (MCC: -0.01, balanced accuracy: 0.5), but do not outperform the Dummy Stratified Classifier significantly. Again, by definition, the performance of the Dummy Minority Classifier and the Dummy Majority Classifier remains unchanged (MCC: 0, balanced accuracy: 0.5), whereas the Dummy Uniform Classifier achieves an MCC of -0.04 and a balanced accuracy of 0.49. However, three classifiers based on LIWC scores significantly outperform the corresponding classifier based on Big Five scores, namely the Support Vector Classifier (p=0.007), the Decision Tree Classifier (p=0.049), and the Random Forest Classifier (p=0.025).

Figure 26 visualizes the MCC of the pre-trained German BERT model during the training (left) and the test phase (right), respectively, including the MCC of the corresponding Dummy Stratified Classifiers (depicted using the dashed lines). As Figure 26 shows, the performance on the training data converges to an MCC of 1 after four epochs in all five iterations of the cross-validation loop, resulting in over-fitted models after the second epoch. After the second epoch, the MCCs of the five folds of the test data range between 0.22 (weak positive relationship) and 0.44 (moderate positive relationship). A more detailed analysis of the confusion matrices reveals that the BERT models achieve a high specificity (ranging between 0.98 and 1.00) and, therefore, are able to identify the absence of salient incentives to fake reliably. However, the low sensitivity (ranging between 0.06 and 0.31) indicates that the BERT models fail to reliably identify all participants who had incentives to fake their cooperativeness. After aggregating over the five folds, these models significantly outperform corresponding Dummy Stratified Classifiers (MCCs ranging between -0.14 and 0.17), as McNemar's test shows ($p<0.01$ for all epochs).

In the absence of salient incentives to fake one's personality, both the classifiers based on Big Five scores and the linguistic scores extracted from written self-descriptions are suitable to assess subject's cooperativeness. However, job applications typically resemble a situation where applicants have an incentive to present themselves as cooperative as possible (Birkeland et al., 2006; Rosse

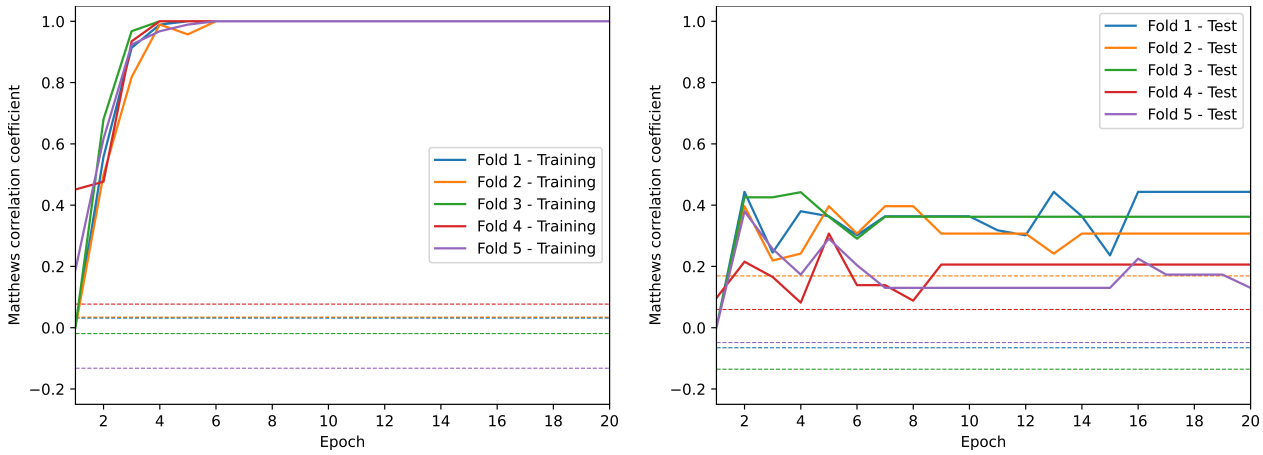


Figure 26: MCCs for predictions of incentives to fake based on the raw text of written self-descriptions (training and test performance)

et al., 1998). In the presence of such incentives, we observe that both the Big Five scores and the linguistic scores lose predictive power. The decrease in predictive power is especially high for the Big Five scores, which suggests that written self-descriptions are less vulnerable to faking than self-reported psychometric tests.

RA 6 addresses this phenomenon with a controlled online experiment with 400 participants. We first elicited the participants' *true* cooperativeness in a public-goods game, then collected a written self-description in the spirit of a cover letter and responses to a 10-item Big-Five questionnaire, and finally introduced an exogenous incentive to fake being cooperative for one quarter of the participants. Six standard classifiers were trained on the Baseline data and evaluated out-of-sample and out-of-context, while a fine-tuned German BERT model tested whether the presence of faking incentives can be detected directly from raw text. The results show that linguistic classifiers retain their predictive power compared to psychometric classifiers once salient incentives to fake arise. Furthermore, a pre-trained language model reliably identifies texts written under those incentives. The contribution is again threefold. First, to the best of our knowledge, RA 6 is the first paper to report empirical evidence on how NLP-based assessments perform relative to psychometric tests, how incentives to fake affect the performance of AI-driven personality assessments, and whether AI can detect the presence of incentives to fake in natural language. Second, we investigate these questions in the context of a real-world problem since job applications resemble a situation in which applicants have incentives to fake being cooperative. Our results suggest that self-reported personality tests are not suitable to assess applicants' cooperativeness, whereas cover letters offer untapped potential for an automated assessment of cooperativeness. Third, our interdisciplinary study bridges the gap between machine learning and experimental economics. In conclusion, RA 6 extends AI-supported process monitoring from customer interactions to human resource processes in the BPM lifecycle through rigorous experimental evaluation.

5 Conclusion

5.1 Summary

Across the BPM lifecycle, from discovery to analysis and redesign to monitoring, AI and especially GenAI are unlocking unprecedented opportunities to enhance BPI. This thesis has shown that organizations can overcome three traditional pain points by augmenting these phases with AI. First, organizations can achieve full transparency of the as-is process by mining unstructured textual data. Second, they can generate and evaluate process improvement ideas using LLMs guided by knowledge and data. Third, they can automate processes with AI through chatbots and AI agents while maintaining oversight through process-aware monitoring.

Addressing the first research objective, this thesis develops techniques to extract event logs from unstructured textual data at scale, thus revealing the situations where processes deviate from their intended sequence and where improvement opportunities arise. For example, RA 1 introduces an NLI-based pipeline that takes customer service conversations and identifies the discussed topics and conducted process activities to represent them in an XES event log. By leveraging a pretrained language model with carefully designed hypotheses, this approach can accurately classify each message in a conversation into activity categories without requiring large annotated training sets. The result is a high-quality event log of customer support interactions, which can then be fed into any process mining tool for discovery. We demonstrated the approach on real-world datasets from three companies' Twitter support accounts, showing that even with as few as 300 labeled samples for fine-tuning, the NLI method outperforms traditional text-classification baselines in accurately extracting topics and process activities.

RA 2 extends this idea by recognizing that many processes are naturally object-centric, i. e., multiple business objects interact not only within, but across multiple process instances. Traditional case-centric event logs have limitations in representing such complex relationships. RA 2 presents an approach to extract OCELs from textual data using a two-stage pipeline consisting of a collector and a refiner. The collector reads textual descriptions and identifies all events with their associated objects, whereas the refiner then cleans and structures this preliminary OCEL to ensure that relationships between events and objects are correctly captured. We instantiate this pipeline in four variants by combining heuristic NLP and GenAI techniques for each of the two stages. The configurations with a generative collector achieve the highest extraction quality, with the fully generative variant in particular producing coherent and standardized event and object labels. Both RA 1 and RA 2 contribute to building a more comprehensive as-is process model by incorporating previously untapped data sources, laying the foundation for better analysis and improvement in later phases. These studies demonstrate that with AI, specifically NLP and GenAI, organizations can significantly broaden their process visibility into sources like conversations and documents, thereby ensuring no relevant behavior is overlooked when seeking improvements.

With a clear view of the as-is process, the following phases in the BPM lifecycle are process analysis (diagnosing weaknesses and their impact) and redesign (developing a better-performing to-be process model), which are addressed by the second research objective of this thesis. To provide AI-driven support for these phases, we developed two complementary artifacts. RA 3 introduces the Process Improvement Copilot, a RAG-enhanced LLM-based PIIS to support the generation of BPI ideas. Given one or multiple – either automatically identified or manually provided – inefficiencies in a process, the Process Improvement Copilot uses an LLM to generate tailored suggestions for overcoming these issues. The crucial innovation is that it employs RAG on a knowledge base of BPI patterns and case studies to ensure that the suggestions are not just hallucinated by the LLM, but build upon proven best practices. We evaluated the Process Improvement Copilot through multiple evaluation activities, including expert interviews and a BPI workshop with a multinational technology conglomerate. The feedback was encouraging, as the majority of the participants found the tool useful, easy to use, and time-saving. Even ideas that were not immediately applicable were not completely off-base, which is a significant achievement for trust in such a system and indicates that an AI-based assistant, if properly guided by retrieval of trusted knowledge, can indeed augment human creativity in BPI initiatives.

RA 4 presents ABuPrOpt, an LLM-driven tool that takes an existing process model and a corresponding event log, and produces an improved to-be model, guided by explicit improvement objectives. ABuPrOpt manipulates the control flow of the process model, essentially the sequence of activities, and then uses a simulation to evaluate the new process model quantitatively. The LLM's computational creativity allows ABuPrOpt to explore a wide solution space, including adding entirely new activities or variants that were neither present in the original process nor the corresponding event log. Furthermore, the feasibility of the proposed changes is ensured as the LLM implicitly considers process dependencies from the input process model. Once a candidate of an improved process model is generated, ABuPrOpt estimates the cycle time and execution cost of both the existing and the suggested process model. Our evaluation with five public datasets showed that ABuPrOpt was able to produce improved models that outperformed the original counterparts on the given objectives. Moreover, all generated models were sound and feasible, thus respecting constraints one would expect in the respective domains. From a practical perspective, ABuPrOpt is an ideation tool that can produce and evaluate a set of to-be process model candidates for consideration, as it reduces the manual effort to brainstorm and assess each alternative. With ABuPrOpt's ability to incorporate custom objectives, organizations can tailor the optimization to their individual needs rather than generic objectives, such as time and cost. Ultimately, RA 4 demonstrates that GenAI can go beyond suggesting ideas as it creates executable process models, thus taking BPI automation to a new level. The study also emphasizes that combining GenAI with analytical techniques is crucial, enabling the human experts to remain in the loop by choosing from the AI-generated suggestions based on the quantitative evidence provided. Together, RA 3 and RA 4 demonstrate how AI in the analysis and redesign phases of the BPM lifecycle can not only uncover weaknesses in business processes, but also generate and assess ideas on how the business process at hand should be changed.

As organizations embrace AI to automate operational work through chatbots or AI agents, a new set of challenges arises in the process monitoring phase. In line with the third research objective, this thesis advances AI-supported process monitoring techniques, specifically for scenarios where processes are augmented or operated by AI, to ensure conformance and performance standards are met. To this end, RA 5 introduces a novel approach to quantify a chatbot's adherence to business processes. We extend the typical machine learning evaluation workflow using established conformance checking metrics from process mining. Our approach leverages the NLI-based event log extraction pipeline from RA 1 to convert human-to-human conversation transcripts into event logs, which represent the intended process. Next, we trained a chatbot on these conversations and let it converse on unseen dialogues, and similarly captured those human-to-chatbot conversations as event logs. Given these two sets of traces, one from the expected process and one from the chatbot's actual behavior, we can quantify how well the chatbot's sequence of actions fits the patterns of the underlying business process. As part of the evaluation, we also demonstrated comparing the chatbot's traces to a normative process model. In practice, this evaluation approach is crucial before deploying a chatbot to customer-facing environments. Therefore, RA 5's approach helps organizations select and fine-tune chatbot models that are linguistically capable and process-compliant, thus contributing to a safer and more effective automation of customer service processes.

In a different vein, RA 6 addresses process monitoring in the context of recruitment processes, focusing on the performance and fairness of AI-based evaluations. We set up a controlled experiment to investigate whether an NLP-based approach could more reliably predict applicants' true cooperativeness than traditional psychometric questionnaires. Having recruited 400 participants, we elicited their true cooperative behavior via a public goods game. We then divided the participants into groups. One group was told their payment in the experiment would depend on appearing cooperative (making incentives to fake salient), whereas the control group had no such incentive. Each participant had to complete two assessments. First, writing a 3,000-character self-description as if applying for a job, and second, filling out a short Big Five personality test. Then we trained two sets of machine learning models, one set based on the text data using linguistic features from the cover letters, and one set of models based on the answers to the personality test, to predict the true cooperativeness elicited through the public goods game. When no salient incentives to fake being cooperative were present, both sets of models had some predictive power. However, once a salient incentive to fake being cooperative was introduced, the models' accuracy based on the personality tests declined significantly, whereas the NLP-based models retained their predictive power. Moreover, we found that a pre-trained language model could reliably detect signs of faking in the writing. These findings suggest that AI can be used to assess personality traits in a richer way and potentially flag candidates portraying themselves deceptively. Hence, organizations could integrate AI-driven analysis into their recruitment process to get more reliable indicators of the applicants' personality, even when candidates are strategically altering their responses. In summary, RA 5 and RA 6 demonstrate how to ensure conformance to process rules and effectiveness of outcomes when organizations automate parts of processes through AI.

5.2 Limitations

The contributions of this thesis, as outlined in the previous sections, should be considered in light of some limitations. While each of the six RAs exhibits individual limitations, the following paragraphs highlight three overarching limitations of the thesis as a whole.

First, the techniques and LLMs used in the RAs, though contemporary at the time of development, have quickly aged due to the fast-paced advances in AI. For instance, RA 3 and RA 4 rely on OpenAI's GPT-4o released in May 2024, and RA 6 extracts linguistic features from written self-descriptions using the LIWC dictionary. At the time of writing this paragraph, GPT-5 is OpenAI's leading LLM, and even though it is difficult to tell from the two model names, OpenAI released six more generations (o1, o3, GPT-4.5, GPT-4.1, o4-mini, GPT-OSS) of foundation models in between. The more recent LLMs have significantly improved in understanding context and semantics, which means the developed approaches should be evaluated again with these LLMs. Likewise, the LIWC scores in RA 6 may miss subtle language cues that state-of-the-art LLMs can capture. Therefore, some approaches in the thesis, such as LIWC-based text analysis or earlier-generation LLMs might not fully leverage the richer semantic representations offered by current AI techniques. Future work will need to update these components to harness the latest models, which are more powerful and capable of capturing nuances beyond the scope of older approaches.

Second, several artifacts and methods developed in this thesis were evaluated in controlled settings or with historical datasets, rather than through real-world deployment. RA 1 and RA 2, for example, introduce novel pipelines for constructing event logs from textual data, but these approaches were tested on existing public datasets and synthetic cases rather than in an ongoing business process in industry. RA 4 was evaluated using academic datasets, and the experiment using cover letters to predict cooperativeness in RA 6 was conducted with recruited participants in a controlled scenario. While these evaluations provide initial evidence of the effectiveness of the developed approaches, they fall short of demonstrating impact in organizational settings. In contrast, RA 3 included a business process improvement workshop at a multinational technology conglomerate, which provided valuable feedback from actual users. Aside from that case, the thesis lacks real-world evaluation showing how the developed artifacts and methods solve business problems over time, thus limiting our understanding of how the proposed solutions perform in practice. Future research should aim for more studies deploying these AI-based BPI tools in organizational settings and assess practical considerations like reliability over time and measurable business impact.

Third, several of the GenAI-driven components, notably in RA 3 and RA 4, exhibit a non-negligible proportion of suggestions or outputs that are irrelevant, ineffective, or otherwise not useful. In the process improvement workshop of RA 3, the participants rated only approximately 20% of the AI-generated improvement ideas as relevant and about 40% of the suggested follow-up steps as actionable. Similarly, the generated to-be process models in RA 4 did not always yield improvements. Out of 25 iterative improvement cycles attempted on five different process models, only 12 resulted in an actual reduction of the cycle time. Such behavior is not unexpected, as current

LLMs can produce plausible-sounding outputs that are not guaranteed to be optimal or even correct. However, it is important to put this limitation into perspective. First, if a process model has very cryptic or ambiguous activity names, such as the process model derived from the Sepsis Cases event log in RA 4, neither an LLM nor a human expert can suggest improvements because the process model lacks clarity. Second, the cost of generating suggestions is relatively low. Even using the most advanced (and typically expensive) LLMs available, producing a large number of ideas or model variants is cheap and fast compared to the cost of a workshop with human experts. Even if only a few suggestions are of high quality, the less useful ones can be filtered out with minimal effort. Indeed, the feedback from the workshop in RA 3 indicated that having just a few high-quality improvement ideas is far more valuable than a long list of mediocre ideas. Third, the rapid development of LLMs suggests that this issue of low-quality outputs can be mitigated over time. Newer LLMs are more adept at understanding context and following user instructions, and they tend to produce more relevant and coherent suggestions. In summary, while the GenAI-based artifacts in this thesis sometimes produce suboptimal ideas or redesigns, this limitation can be managed by combining computational creativity with human judgment, and it is likely to be diminished as both the data and the models improve going forward.

5.3 Future Work

Building on the findings and limitations of this thesis, the following paragraphs outline three promising directions for future research. These directions aim to address the limitations discussed above and extend the contributions of this thesis.

The first step is to integrate the complementary approaches from RA 1, RA 2, RA 3, and RA 4 into a holistic PIIS. In this thesis, RA 1 and RA 2 developed methods to construct event logs from textual data, thus enhancing transparency of the as-is process, RA 3 introduced an LLM-driven copilot for generating process improvement ideas from identified inefficiencies, and RA 4 demonstrated the automated generation and evaluation of optimized process models. Merging these components could yield a comprehensive pipeline that starts from raw textual data and ends with validated to-be process models. For example, an OCEL extracted using the approach from RA 2 could directly be fed into an AI assistant (as in RA 3), which diagnoses inefficiencies and suggests improvements, and those suggestions are automatically translated into improved process models (as in RA 4) for simulation and evaluation. Such an integrated system would essentially function as a next-generation, AI-driven PIIS covering discovery, analysis, and redesign in a continuous loop. While there are practical challenges to this integration, such as ensuring data formats and interfaces are compatible, and maintaining traceability from an inefficiency to an implemented change, the potential gains in BPI efficiency are considerable, as a unified tool could significantly accelerate continuous BPI by automating not only isolated tasks but the entire improvement cycle.

A second important avenue is to enhance the process analysis phase by leveraging AI for more intelligent and automated detection of process inefficiencies. Although crucial for the targeted development of process improvement ideas, inefficiency identification was researched to a limited

extent in this thesis. For instance, the prototype in RA 3 used three predefined inefficiency detection patterns to trigger improvement suggestions. Future research should develop methods for AI-based inefficiency detection that can analyze process execution data and pinpoint issues without relying on predefined patterns. One idea is to employ AI agents to examine event logs and identify anomalies and bottlenecks by recognizing complex patterns beyond traditional algorithms. These agents could be prompted with a structured summary of a process's cases and asked to hypothesize what might be going wrong. With the most recent generation of LLMs, it is conceivable that AI agents can suggest inefficiencies and corresponding root causes based on data patterns and even textual context from incident logs or customer feedback. Integrating root cause analysis techniques, such as causal inference or process query approaches, into these agents would further strengthen their diagnostic capabilities. Once performance issues are detected (e. g., above-average cycle times for certain cases), the system can interactively drill down into the data to find contributing factors like resource bottlenecks or external dependencies. Therefore, a GenAI-based copilot with a clear understanding of the root cause of a problem can propose much more effective and targeted improvement ideas, rather than generic suggestions that address only symptoms. In summary, future research should strive to make AI not just a tool for generating solutions, but also for discovering problems, thus ultimately enabling a fully AI-driven analysis phase in the BPM lifecycle.

To support the above two future directions and facilitate the integration of AI agents with process mining tools and enterprise systems, we propose exploring the use of the emerging Model Context Protocol (MCP). The MCP standardizes the communication between AI agents and external systems by exposing system functionalities as modular tools through defined interfaces. In essence, the MCP allows an LLM-based AI agent to call procedures on enterprise systems in a secure and interoperable way, rather than operating in an isolated system. For example, using MCP, an AI agent could automatically query a process mining tool for the event log of a particular process, perform conformance or performance analyses, retrieve relevant fragments of a process model, and execute a process simulation through standardized interfaces that the LLM can invoke. This promises not only easier integration and scaling of the research artifacts of this thesis, but also greater extensibility, as new tools for automated root cause analysis can be added into the architecture without redesigning the entire system. Adopting the MCP in future implementations would also move the research closer to real-world deployment, since enterprises are likely to favor solutions built on standard protocols for maintainability and security.

In summary, addressing the above future work directions can broaden and deepen the contributions of this thesis. By combining the artifacts for the discovery, analysis, and redesign phases into a holistic PIIS, enhancing AI's role in diagnosing process inefficiencies, and utilizing protocols like MCP to integrate with enterprise systems, future research can create more powerful AI-driven BPI solutions. These steps will ensure that the next generation of PIISs is academically engaging and practically impactful in driving continuous process improvement in organizations.

Use of Writing Assistance

Please note that I have utilized various writing-assistance tools, such as Grammarly, DeepL, and ChatGPT to enhance the language and readability of this thesis. Of course, I take full responsibility for the content and have thoroughly reviewed and edited the material as necessary.

References

Adamopoulou, E., Moussiades, L. (2020). Chatbots: History, technology, and applications. *Machine Learning with Applications* 2, 100006. doi:10.1016/j.mlwa.2020.100006.

Afflerbach, P., Hohendorf, M., Manderscheid, J. (2017). Design it like Darwin – A value-based application of evolutionary algorithms for proper and unambiguous business process redesign. *Information Systems Frontiers* 19 (5), 1101–1121. doi:10.1007/s10796-016-9715-1.

Agrawal, G., Kumarage, T., Alghamdi, Z., Liu, H. (2024). Can Knowledge Graphs Reduce Hallucinations in LLMs? : A Survey, in: Duh, K., Gomez, H., Bethard, S. (Eds.), *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, Association for Computational Linguistics, Mexico City, Mexico. pp. 3947–3960. doi:10.18653/v1/2024.naacl-long.219.

Alter, S. (2014). Theory of Workarounds. *Communications of the Association for Information Systems* 34, 1041–1066. doi:10.17705/1CAIS.03455.

Al-Mashari, M., Zairi, M. (1999). BPR implementation process: an analysis of key success and failure factors. *Business Process Management Journal* 5 (1), 87–112. doi:10.1108/14637159910249108.

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

Androutsopoulou, A., Karacapilidis, N., Loukis, E., Charalabidis, Y. (2019). Transforming the communication between citizens and government through AI-guided chatbots. *Government Information Quarterly* 36 (2), 358–367. doi:10.1016/j.giq.2018.10.001.

Bader, M., Antony, J., Jayaraman, R., Swarnakar, V., Goonetilleke, R.S., Maalouf, M., Garza-Reyes, J.A., Linderman, K. (2023). Why do process improvement projects fail in organizations? A review and future research agenda. *International Journal of Lean Six Sigma* 15 (3), 664–690. doi:10.1108/IJLSS-07-2023-0126.

Balaguer, A., Benara, V., de Freitas Cunha, R.L., de M. Estevão Filho, R., Hendry, T., Holstein, D., Marsman, J., Mecklenburg, N., Malvar, S., Nunes, L.O., Padilha, R., Sharp, M., Silva, B., Sharma, S., Aski, V., Chandra, R. (2024). RAG vs Fine-tuning: Pipelines, Tradeoffs, and a Case Study on Agriculture. arXiv:2401.08406.

Banziger, R.B., Basukoski, A., Chaussalet, T. (2018). Discovering Business Processes in CRM Systems by Leveraging Unstructured Text Data, in: *2018 IEEE 20th International Conference on High Performance Computing and Communications; IEEE 16th International Conference on Smart City; IEEE 4th International Conference on Data Science and Systems (HPCC/SmartCity/DSS)*, Exeter, UK. pp. 1571–1577. doi:10.1109/HPCC/SmartCity/DSS.2018.00257.

Beerepoot, I., Di Ciccio, C., Reijers, H.A., Rinderle-Ma, S., Bandara, W., Burattin, A., Calvanese, D., Chen, T., Cohen, I., Depaire, B., Di Federico, G., Dumas, M., van Dun, C., Fehrer, T., Fischer, D.A., Gal, A., Indulska, M., Isahagian, V., Klinkmüller, C., Kratsch, W., Leopold, H., Van Looy, A., Lopez, H., Lukumbuzya, S., Mendling, J., Meyers, L., Moder, L., Montali, M., Muthusamy, V., Reichert, M., Rizk, Y., Rosemann, M., Röglinger, M., Sadiq, S., Seiger, R., Slaats, T., Simkus, M., Someh, I.A., Weber, B., Weber, I., Weske, M., Zerbato, F. (2023). The biggest business process management problems to solve before we die. *Computers in Industry* 146, 103837. doi:10.1016/j.compind.2022.103837.

Beerepoot, I., van de Weerd, I., Reijers, H.A. (2019). Business Process Improvement Activities: Differences in Organizational Size, Culture, and Resources, in: Hildebrandt, T., van Dongen, B.F., Röglinger, M., Mendling, J. (Eds.), *Business Process Management. BPM 2019*, Vienna, Austria: Springer. pp. 402–418. doi:10.1007/978-3-030-26619-6_26.

Beheshti, A., Yang, J., Sheng, Q.Z., Benatallah, B., Casati, F., Dustdar, S., Nezhad, H.R.M., Zhang, X., Xue, S. (2023). ProcessGPT: Transforming Business Process Management with Generative Artificial Intelligence, in: *2023 IEEE International Conference on Web Services (ICWS)*, Los Alamitos, CA, USA: IEEE. pp. 731–739. doi:10.1109/ICWS60048.2023.00099.

Berti, A. (2023). Collection of Object-Centric Event Logs (OCEL 2.0 format; JSON specification). doi:10.5281/zenodo.8433706.

Berti, A., Koren, I., Adams, J.N., Park, G., Knopp, B., Graves, N., Rafiei, M., Liß, L., Tacke Genannt Unterberg, L., Zhang, Y., Schwanen, C., Pegoraro, M., van der Aalst, W.M.P. (2024). OCEL (Object-Centric Event Log) 2.0 Specification. arXiv:2403.01975.

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.

Birkeland, S.A., Manson, T.M., Kisamore, J.L., Brannick, M.T., Smith, M.A. (2006). A Meta-Analytic Investigation of Job Applicant Faking on Personality Measures. *International Journal of Selection and Assessment* 14 (4), 317–335. doi:10.1111/j.1468-2389.2006.00354.x.

Blocker, C.P., Flint, D.J., Myers, M.B., Slater, S.F. (2011). Proactive customer orientation and its role for creating customer value in global markets. *Journal of the Academy of Marketing Science* 39 (2), 216–233. doi:10.1007/s11747-010-0202-9.

Bond, G.D., Lee, A.Y. (2005). Language of lies in prison: Linguistic classification of prisoners' truthful and deceptive natural language. *Applied Cognitive Psychology* 19 (3), 313–329. doi:10.1002/acp.1087.

Boyd, R.L., Pennebaker, J.W. (2017). Language-based personality: a new approach to personality in a digital world. *Current Opinion in Behavioral Sciences* 18, 63–68. doi:10.1016/j.cobeha.2017.07.017.

Brandtzaeg, P.B., Følstad, A. (2017). Why People Use Chatbots, in: Kompatsiaris, I., Cave, J., Satsiou, A., Carle, G., Passani, A., Kontopoulos, E., Diplaris, S., McMillan, D. (Eds.), *Internet Science. INSCI 2017*, Springer International Publishing, Cham. pp. 377–392. doi:10.1007/978-3-319-70284-1_30.

Brown, T.B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D.M., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., Amodei, D. (2020). Language Models are Few-Shot Learners, in: *Advances in Neural Information Processing Systems (NeurIPS 2020)*, Online. pp. 1877–1901.

Buijs, J.C.A.M., van Dongen, B.F., van der Aalst, W.M.P. (2012). On the Role of Fitness, Precision, Generalization and Simplicity in Process Discovery, in: Meersman, R., Panetto, H., Dillon, T., Rinderle-Ma, S., Dadam, P., Zhou, X., Pearson, S., Ferscha, A., Bergamaschi, S., Cruz, I.F. (Eds.), *On the Move to Meaningful Internet Systems. OTM 2012*, Rome, Italy: Springer, Berlin, Heidelberg. pp. 305–322. doi:10.1007/978-3-642-33606-5_19.

Buss, A., Kecht, C., Kratsch, W., Röglinger, M., Sadeghianasl, S., Wynn, M.T. (2025). From Words to Workflows: Extracting Object-Centric Event Logs from Textual Data, in: Pufahl, L., Rosenthal, K., España, S., Nurcan, S. (Eds.), *Intelligent Information Systems. CAiSE 2025*, Vienna, Austria: Springer, Cham. pp. 37–44. doi:10.1007/978-3-031-94590-8_5.

Buss, A., Kecht, C., Kratsch, W., Röglinger, M., Sadeghianasl, S., Wynn, M.T. (2026). Process mining between the lines: Extracting object-centric event logs from textual data. *Information Systems* 140, 102713. doi:10.1016/j.is.2026.102713.

Calvanese, D., Montali, M., Syamsiyah, A., van der Aalst, W.M.P. (2016). Ontology-Driven Extraction of Event Logs from Relational Databases, in: Reichert, M., Reijers, H.A. (Eds.), *Business Process Management Workshops. BPM 2016*, Springer, Cham. pp. 140–153. doi:10.1007/978-3-319-42887-1_12.

Chen, D.L., Schonger, M., Wickens, C. (2016). oTree – An open-source platform for laboratory, online, and field experiments. *Journal of Behavioral and Experimental Finance* 9, 88–97. doi:10.1016/j.jbef.2015.12.001.

Chen, R., Gong, J. (2018). Can self selection create high-performing teams? *Journal of Economic Behavior & Organization* 148, 20–33. doi:10.1016/j.jebo.2018.02.004.

Davenport, T.H. (1993). Process innovation: reengineering work through information technology. Harvard Business School Press, Boston, MA, USA.

Davis, F.D. (1989). Perceived Usefulness, Perceived Ease of Use, and User Acceptance of Information Technology. *MIS Quarterly* 13 (3), 319–340. doi:10.2307/249008.

de Leoni, M., Mannhardt, F. (2015). Road Traffic Fine Management Process. doi:10.4121/uuid:270fd440-1057-4fb9-89a9-b699b47990f5.

Derczynski, L. (2016). Complementarity, F-score, and NLP Evaluation, in: Calzolari, N., Choukri, K., Declerck, T., Goggi, S., Grobelnik, M., Maegaard, B., Mariani, J., Mazo, H., Moreno, A., Odijk, J., Piperidis, S. (Eds.), *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, European Language Resources Association (ELRA), Portorož, Slovenia. pp. 261–266.

Dietterich, T. (1998). Approximate Statistical Tests for Comparing Supervised Classification Learning Algorithms. *Neural Computation* 10 (7), 1895–1923. doi:10.1162/089976698300017197.

Dumas, M., Rosa, M.L., Mendling, J., Reijers, H.A. (2018). Fundamentals of Business Process Management. 2nd ed., Springer, Berlin, Heidelberg. doi:10.1007/978-3-662-56509-4.

Fahland, D., Fournier, F., Limonad, L., Skarbovsky, I., Swevels, A.J.E. (2025). How well can a large language model explain business processes as perceived by users? *Data & Knowledge Engineering* 157, 102416. doi:10.1016/j.datak.2025.102416.

Falk, T., Griesberger, P., Johannsen, F., Leist, S. (2013). Patterns For Business Process Improvement – A First Approach, in: *Proceedings of the 21st European Conference on Information Systems (ECIS 2013)*, Utrecht, The Netherlands.

Fehrer, T., Fischer, D.A., Leemans, S.J.J., Röglinger, M., Wynn, M.T. (2022). An assisted approach to business process redesign. *Decision Support Systems* 156, 113749. doi:10.1016/j.dss.2022.113749.

Fehrer, T., Moder, L., Röglinger, M. (2025). A Taxonomy for Process Improvement and Innovation Systems. *Business & Information Systems Engineering* doi:10.1007/s12599-025-00928-4.

Fellmann, M., Hogrebe, F., Thomas, O., Nüttgens, M. (2011). Checking the Semantic Correctness of Process Models – An Ontology-driven Approach Using Domain Knowledge and Rules. *Enterprise Modelling and Information Systems Architectures* 6 (3), 25–35. doi:10.18417/emisa.6.3.2.

Feuerriegel, S., Hartmann, J., Janiesch, C., Zschech, P. (2024). Generative AI. *Business & Information Systems Engineering* 66 (1), 111–126. doi:10.1007/s12599-023-00834-7.

Figl, K., Recker, J. (2016). Process innovation as creative problem solving: An experimental study of textual descriptions and diagrams. *Information & Management* 53 (6), 767–786. doi:10.1016/j.im.2016.02.008.

Fischbacher, U., Gächter, S., Fehr, E. (2001). Are people conditionally cooperative? Evidence from a public goods experiment. *Economics Letters* 71 (3), 397–404. doi:10.1016/S0165-1765(01)00394-9.

Følstad, A., Brandtzæg, P.B. (2017). Chatbots and the new world of HCI. *Interactions* 24 (4), 38–42. doi:10.1145/3085558.

Geeganage, D.T.K., Wynn, M.T., ter Hofstede, A.H. (2022). Text2EL: Exploiting Unstructured Text for Event Log Enrichment, in: *2022 16th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS)*, Dijon, France: IEEE. pp. 1–8. doi:10.1109/SITIS57111.2022.00010.

Goldberg, L.R. (1990). An alternative "description of personality": The Big-Five factor structure. *Journal of Personality and Social Psychology* 59 (6), 1216–1229. doi:10.1037/0022-3514.59.6.1216.

Gosling, S.D., Rentfrow, P.J., Swann, W.B. (2003). A very brief measure of the Big-Five personality domains. *Journal of Research in Personality* 37 (6), 504–528. doi:10.1016/S0092-6566(03)00046-1.

Gregor, S., Hevner, A.R. (2013). Positioning and presenting design science research for maximum impact. *MIS Quarterly* 37 (2), 337–355. doi:10.25300/MISQ/2013/37.2.01.

Grisold, T., Mendling, J., Otto, M., vom Brocke, J. (2021). Adoption, use and management of process mining in practice. *Business Process Management Journal* 27 (2), 369–387. doi:10.1108/BPMJ-03-2020-0112.

Groß, S., Grisold, T., Mendling, J., Haase, J. (2024). Idea generation in exploitative and explorative business process redesign techniques. *Information Systems and e-Business Management* 22 (3), 527–555. doi:10.1007/s10257-024-00684-0.

Gross, S., Malinova, M., Mendling, J. (2019). Navigating Through the Maze of Business Process Change Methods, in: Bui, T. (Ed.), *Proceedings of the 52nd Hawaii International Conference on System Sciences. HICSS 2019*, Grand Wailea, Maui, HI, USA. pp. 6270–6279. doi:10.24251/HICSS.2019.754.

Gross, S., Stelzl, K., Grisold, T., Mendling, J., Röglinger, M., vom Brocke, J. (2021). The Business Process Design Space for exploring process redesign alternatives. *Business Process Management Journal* 27 (8), 25–56. doi:10.1108/BPMJ-03-2020-0116.

Gunson, N., Marshall, D., Morton, H., Jack, M. (2011). User perceptions of security and usability of single-factor and two-factor authentication in automated telephone banking. *Computers & Security* 30 (4), 208–220. doi:10.1016/j.cose.2010.12.001.

Günther, C.W., Rozinat, A. (2012). Disco: Discover Your Processes, in: Lohmann, N., Moser, S. (Eds.), *Proceedings of the Demonstration Track of the 10th International Conference on Business Process Management. BPM 2012*, Tallinn, Estonia. pp. 40–44.

Hammer, M., Champy, J. (1993). Reengineering the Corporation: A Manifesto for Business Revolution. 1st ed., HarperCollins, New York.

Hancock, J.T., Curry, L.E., Goorha, S., Woodworth, M. (2007). On Lying and Being Lied To: A Linguistic Analysis of Deception in Computer-Mediated Communication. *Discourse Processes* 45 (1), 1–23. doi:10.1080/01638530701739181.

Hauch, V., Blandón-Gitlin, I., Masip, J., Sporer, S.L. (2015). Are Computers Effective Lie Detectors? A Meta-Analysis of Linguistic Cues to Deception. *Personality and Social Psychology Review* 19 (4), 307–342. doi:10.1177/1088868314556539.

Heinisch, M., Graves, N., van der Aalst, W.M.P. (2024). sOCEL 2.0: A Sustainability-Enriched OCEL of a Hinge Production Process. doi:10.5281/zenodo.13638681.

Hoesch-Klohe, K., Ghose, A., Lê, L.S. (2010). Towards green business process management, in: *2010 IEEE International Conference on Services Computing*, Miami, FL, USA: IEEE. pp. 386–393. doi:10.1109/SCC.2010.21.

Hosseini, S., Merz, M., Röglinger, M., Wenninger, A. (2018). Mindfully going omni-channel: An economic decision model for evaluating omni-channel strategies. *Decision Support Systems* 109, 74–88. doi:10.1016/j.dss.2018.01.010.

Huang, L., Yu, W., Ma, W., Zhong, W., Feng, Z., Wang, H., Chen, Q., Peng, W., Feng, X., Qin, B., Liu, T. (2025). A Survey on Hallucination in Large Language Models: Principles, Taxonomy, Challenges, and Open Questions. *ACM Transactions on Information Systems* 43 (2). doi:10.1145/3703155.

Huang, S.Y., Lee, C.H., Chiu, A.A., Yen, D.C. (2015). How business process reengineering affects information technology investment and employee performance under different performance measurement. *Information Systems Frontiers* 17 (5), 1133–1144. doi:10.1007/s10796-014-9487-4.

IEEE (2016). IEEE Standard for eXtensible Event Stream (XES) for Achieving Interoperability in Event Logs and Event Streams. doi:10.1109/IEEESTD.2016.7740858.

Isaac, R.M., Walker, J.M. (1988). Group Size Effects in Public Goods Provision: The Voluntary Contributions Mechanism. *The Quarterly Journal of Economics* 103 (1), 179–199. doi:10.2307/1882648.

Jansen-Vullers, M.H., Kleingeld, P.A.M., Loosschilder, M.W.N.C., Netjes, M., Reijers, H.A. (2008a). Trade-Offs in the Performance of Workflows – Quantifying the Impact of Best Practices, in: ter Hofstede, A., Benatallah, B., Paik, H.Y. (Eds.), *Business Process Management Workshops. BPM 2007*, Brisbane, Australia: Springer. pp. 108–119. doi:10.1007/978-3-540-78238-4_13.

Jansen-Vullers, M.H., Kleingeld, P.A.M., Netjes, M. (2008b). Quantifying the Performance of Workflows. *Information Systems Management* 25 (4), 332–343. doi:10.1080/10580530802384589.

Jlailaty, D., Grigori, D., Belhajjame, K. (2017). Mining Business Process Activities from Email Logs, in: *2017 IEEE International Conference on Cognitive Computing (ICCC)*, Honolulu, HI, USA. pp. 112–119. doi:10.1109/IEEE.ICCC.2017.28.

Joshi, A., Kale, S., Chandel, S., Pal, D.K. (2015). Likert Scale: Explored and Explained. *Current Journal of Applied Science and Technology* 7 (4), 396–403. doi:10.9734/BJAST/2015/14975.

Joulin, A., Grave, E., Bojanowski, P., Douze, M., Jégou, H., Mikolov, T. (2016). FastText.zip: Compressing text classification models. arXiv:1612.03651.

Joulin, A., Grave, E., Bojanowski, P., Mikolov, T. (2017). Bag of Tricks for Efficient Text Classification, in: Lapata, M., Blunsom, P., Koller, A. (Eds.), *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, Valencia, Spain. pp. 427–431.

Kagel, J., McGee, P. (2014). Personality and cooperation in finitely repeated prisoner's dilemma games. *Economics Letters* 124 (2), 274–277. doi:10.1016/j.econlet.2014.05.034.

Kecht, C., Egger, A., Kratsch, W., Röglinger, M. (2021). Event Log Construction from Customer Service Conversations Using Natural Language Inference, in: *2021 3rd International Conference on Process Mining (ICPM)*, Eindhoven, Netherlands: IEEE. pp. 144–151. doi:10.1109/ICPM53251.2021.9576869.

Kecht, C., Egger, A., Kratsch, W., Röglinger, M. (2023). Quantifying chatbots' ability to learn business processes. *Information Systems* 113, 102176. doi:10.1016/j.is.2023.102176.

Kecht, C., Kurschilgen, M., Strobel, M. (2022). Revival of the Cover Letter? Experimental Evidence on the Performance of AI-driven Personality Assessments, in: Bjørn-Andersen, N., Beck, R., Petter, S., Jensen, T.B., Böhmann, T., Hui, K., Venkatesh, V. (Eds.), *Proceedings of the 43rd International Conference on Information Systems (ICIS 2022)*, Copenhagen, Denmark: Association for Information Systems.

Kerpedzhiev, G.D., König, U.M., Röglinger, M., Rosemann, M. (2021). An Exploration into Future Business Process Management Capabilities in View of Digitalization. *Business & Information Systems Engineering* 63 (2), 83–96. doi:10.1007/s12599-020-00637-0.

Kettinger, W.J., Teng, J.T.C., Guha, S. (1997). Business Process Change: A Study of Methodologies, Techniques, and Tools. *MIS Quarterly* 21 (1), 55–80. doi:10.2307/249742.

Knopp, B., Graves, N. (2023). Container Logistics Object-centric Event Log. doi:10.5281/zenodo.8428084.

Knopp, B., van der Aalst, W.M. (2023). Order Management Object-centric Event Log in OCEL 2.0 Standard. doi:10.5281/zenodo.8428112.

Kojima, T., Gu, S.S., Reid, M., Matsuo, Y., Iwasawa, Y. (2022). Large Language Models are Zero-Shot Reasoners, in: Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., Oh, A. (Eds.), *Advances in Neural Information Processing Systems 35 (NeurIPS 2022)*, New Orleans, LA, USA. pp. 22199–22213.

Koole, S.L., Jager, W., van den Berg, A.E., Vlek, C.A.J., Hofstee, W.K.B. (2016). On the Social Nature of Personality: Effects of Extraversion, Agreeableness, and Feedback about Collective Resource Use on Cooperation in a Resource Dilemma. *Personality and Social Psychology Bulletin* 27 (3), 289–301. doi:10.1177/0146167201273003.

Kourani, H., Berti, A., Schuster, D., van der Aalst, W.M.P. (2024). Process Modeling with Large Language Models, in: van der Aa, H., Bork, D., Schmidt, R., Sturm, A. (Eds.), *Enterprise, Business-Process and Information Systems Modeling. BPMDS EMMSAD 2024*, Limassol, Cyprus: Springer. pp. 229–244. doi:10.1007/978-3-031-61007-3_18.

Kourani, H., van Zelst, S.J. (2023). POWL: Partially Ordered Workflow Language, in: Di Francescomarino, C., Burattin, A., Janiesch, C., Sadiq, S. (Eds.), *Business Process Management. BPM 2023*, Utrecht, The Netherlands: Springer. pp. 92–108. doi:10.1007/978-3-031-41620-0_6.

Kratsch, W., König, F., Röglinger, M. (2022). Shedding light on blind spots – Developing a reference architecture to leverage video data for process mining. *Decision Support Systems* 158, 113794. doi:10.1016/j.dss.2022.113794.

Kreuzer, T., Röglinger, M., Rupprecht, L. (2020). Customer-centric prioritization of process improvement projects. *Decision Support Systems* 133, 113286. doi:10.1016/j.dss.2020.113286.

König, F., Egger, A., Kratsch, W., Röglinger, M., Wördehoff, N. (2025). Unstructured Data in Process Mining: A Systematic Literature Review. *ACM Transactions on Management Information Systems* 16 (3). doi:10.1145/3727148.

König, U.M., Linhart, A., Röglinger, M. (2019). Why do business processes deviate? Results from a Delphi study. *Business Research* 12 (2), 425–453. doi:10.1007/s40685-018-0076-0.

Köpke, J., Safan, A. (2025). Efficient LLM-Based Conversational Process Modeling, in: Gdowska, K., Gómez-López, M.T., Rehse, J.R. (Eds.), *Business Process Management Workshops. BPM 2024*, Krakow, Poland: Springer. pp. 259–270. doi:10.1007/978-3-031-78666-2_20.

Lashkevich, K., Milani, F., Danylyshyn, N. (2023). Analysis templates for identifying improvement opportunities with process mining, in: *Proceedings of the 31st European Conference on Information Systems (ECIS 2023)*, Kristiansand, Norway.

Lazear, E.P., Shaw, K.L. (2007). Personnel Economics: The Economist's View of Human Resources. *Journal of Economic Perspectives* 21 (4), 91–114. doi:10.1257/jep.21.4.91.

Leemans, S.J.J., Fahland, D., van der Aalst, W.M.P. (2013). Discovering Block-Structured Process Models from Event Logs - A Constructive Approach, in: Colom, J.M., Desel, J. (Eds.), *Application and Theory of Petri Nets and Concurrency. PETRI NETS 2013*, Springer, Berlin, Heidelberg. pp. 311–329. doi:10.1007/978-3-642-38697-8_17.

Leemans, S.J.J., Fahland, D., van der Aalst, W.M.P. (2014). Discovering Block-Structured Process Models from Incomplete Event Logs, in: Ciardo, G., Kindler, E. (Eds.), *Application and Theory of Petri Nets and Concurrency*, Springer, Cham. pp. 91–110. doi:10.1007/978-3-319-07734-5_6.

Lehnert, M., Linhart, A., Röglinger, M. (2016). Value-based process project portfolio management: integrated planning of BPM capability development and process improvement. *Business Research* 9 (2), 377–419. doi:10.1007/s40685-016-0036-5.

Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., Stoyanov, V., Zettlemoyer, L. (2020a). BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension, in: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Online. pp. 7871–7880. doi:10.18653/v1/2020.acl-main.703.

Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W.t., Rocktäschel, T., Riedel, S., Kiela, D. (2020b). Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks, in: Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., Lin, H. (Eds.), *Advances in Neural Information Processing Systems 33 (NeurIPS 2020)*, Curran Associates, Inc. pp. 9459–9474.

Li, C.Y., Shinde, T., He, W., Lau, S.S.F., Hiew, M.X.B., Tam, N.T.L., Joshi, A., van der Aalst, W.M.P. (2023). Unveiling Bottlenecks in Logistics: A Case Study on Process Mining for Root Cause Identification and Diagnostics in an Air Cargo Terminal, in: Monti, F., Rinderle-Ma, S., Ruiz Cortés, A., Zheng, Z., Mecella, M. (Eds.), *Service-Oriented Computing. ICSOC 2023*, Rome, Italy: Springer. pp. 291–307. doi:10.1007/978-3-031-48424-7_21.

Limam Mansar, S., Reijers, H.A., Ounnar, F. (2009). Development of a decision-making strategy to improve the efficiency of BPR. *Expert Systems with Applications* 36 (2, Part 2), 3248–3262. doi:10.1016/j.eswa.2008.01.008.

Liss, L., Elbert, N., Flath, C.M., van der Aalst, W.M.P. (2024). Object-Centric Event Log for Age of Empires Game Interactions. doi:10.5281/zenodo.13365584.

MacCartney, B., Manning, C.D. (2008). Modeling Semantic Containment and Exclusion in Natural Language Inference, in: *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*, Manchester, UK. pp. 521–528.

Magaldi, D., Berler, M. (2020). Semi-structured Interviews, in: Zeigler-Hill, V., Shackelford, T.K. (Eds.), *Encyclopedia of Personality and Individual Differences*. Springer International Publishing, Cham, pp. 4825–4830. doi:10.1007/978-3-319-24612-3_857.

Malinova, M., Gross, S., Mendling, J. (2022). A study into the contingencies of process improvement methods. *Information Systems* 104, 101880. doi:10.1016/j.is.2021.101880.

Mannhardt, F. (2016). Sepsis Cases - Event Log. doi:10.4121/uuid:915d2bfb-7e84-49ad-a286-dc35f063a460.

Mannhardt, F. (2017). Hospital Billing - Event Log. doi:10.4121/uuid:76c46b83-c930-4798-a1c9-4be94dfeb741.

Maynez, J., Narayan, S., Bohnet, B., McDonald, R. (2020). On Faithfulness and Factuality in Abstractive Summarization, in: Jurafsky, D., Chai, J., Schluter, N., Tetreault, J. (Eds.), *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Online. pp. 1906–1919. doi:10.18653/v1/2020.acl-main.173.

McNemar, Q. (1947). Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika* 12 (2), 153–157. doi:10.1007/BF02295996.

Moder, L., Fehrer, T., Röglinger, M. (2025). Design principles for process improvement and innovation systems. *Business Process Management Journal* doi:10.1108/BPMJ-10-2024-0975.

Morgeson, F.P., Campion, M.A., Dipboye, R.L., Hollenbeck, J.R., Murphy, K., Schmitt, N. (2007). Reconsidering the Use of Personality Tests in Personnel Selection Contexts. *Personnel Psychology* 60 (3), 683–729. doi:10.1111/j.1744-6570.2007.00089.x.

Muñoz-Gama, J., Carmona, J. (2010). A Fresh Look at Precision in Process Conformance, in: Hull, R., Mendling, J., Tai, S. (Eds.), *Business Process Management. BPM 2010*, Springer, Berlin, Heidelberg. pp. 211–226. doi:10.1007/978-3-642-15618-2_16.

Mustansir, A., Shahzad, K., Malik, M.K. (2022). Towards automatic business process redesign: an NLP based approach to extract redesign suggestions. *Automated Software Engineering* 29 (1), 12. doi:10.1007/s10515-021-00316-8.

Newman, M.L., Pennebaker, J.W., Berry, D.S., Richards, J.M. (2003). Lying Words: Predicting Deception from Linguistic Styles. *Personality and Social Psychology Bulletin* 29 (5), 665–675. doi:10.1177/0146167203029005010.

Niedermann, F., Schwarz, H. (2011). Deep Business Optimization: Making Business Process Optimization Theory Work in Practice, in: Halpin, T., Nurcan, S., Krogstie, J., Soffer, P., Proper, E., Schmidt, R., Bider, I. (Eds.), *Enterprise, Business-Process and Information Systems Modeling. BPMDS EMMSAD 2011*, London, UK: Springer. pp. 88–102. doi:10.1007/978-3-642-21759-3_7.

Park, G., Tacke genannt Unterberg, L. (2023). Procure-To-Payment (P2P) Object-centric Event Log in OCEL 2.0 Standard. doi:10.5281/zenodo.8412920.

Park, G., van der Aalst, W.M.P. (2022). Action-oriented process mining: bridging the gap between insights and actions. *Progress in Artificial Intelligence* doi:10.1007/s13748-022-00281-7.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E. (2011). Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* 12 (85), 2825–2830.

Peeperkorn, M., Kouwenhoven, T., Brown, D., Jordanous, A. (2024). Is temperature the creativity parameter of large language models? arXiv:2405.00492.

Peffers, K., Tuunanen, T., Rothenberger, M.A., Chatterjee, S. (2007). A Design Science Research Methodology for Information Systems Research. *Journal of Management Information Systems* 24 (3), 45–77. doi:10.2753/MIS0742-1222240302.

Pennebaker, J.W., Booth, R.J., Boyd, R.L., Francis, M.E. (2015). Linguistic Inquiry and Word Count: LIWC2015.

Poddar, A., Donthu, N., Wei, Y. (2009). Web site customer orientations, Web site quality, and purchase intentions: The role of Web site personality. *Journal of Business Research* 62 (4), 441–450. doi:10.1016/j.jbusres.2008.01.036.

Reijers, H., Liman Mansar, S. (2005). Best practices in business process redesign: an overview and qualitative evaluation of successful redesign heuristics. *Omega* 33 (4), 283–306. doi:10.1016/j.omega.2004.04.012.

Rich, N., Bateman, N. (2003). Companies' perceptions of inhibitors and enablers for process improvement activities. *International Journal of Operations & Production Management* 23 (2), 185–199. doi:10.1108/01443570310458447.

Rinderle, S., Reichert, M. (2006). Data–Driven Process Control and Exception Handling in Process Management Systems, in: Dubois, E., Pohl, K. (Eds.), *Advanced Information Systems Engineering. CAiSE 2006*, Springer, Berlin, Heidelberg. pp. 273–287. doi:10.1007/11767138_19.

Rogge-Solti, A., Senderovich, A., Weidlich, M., Mendling, J., Gal, A. (2016). In Log and Model We Trust? A Generalized Conformance Checking Framework, in: La Rosa, M., Loos, P., Pastor, O. (Eds.), *Business Process Management. BPM 2016*, Springer, Cham. pp. 179–196. doi:10.1007/978-3-319-45348-4_11.

Rosemann, M., vom Brocke, J. (2015). The Six Core Elements of Business Process Management, in: vom Brocke, J., Rosemann, M. (Eds.), *Handbook on Business Process Management 1: Introduction, Methods, and Information Systems*. Springer, Berlin, Heidelberg, pp. 105–122. doi:10.1007/978-3-642-45100-3_5.

Rosse, J.G., Stecher, M.D., Miller, J.L., Levin, R.A. (1998). The impact of response distortion on preemployment personality testing and hiring decisions. *Journal of Applied Psychology* 83 (4), 634–644. doi:10.1037/0021-9010.83.4.634.

Rozinat, A., van der Aalst, W.M.P. (2008). Conformance checking of processes based on monitoring real behavior. *Information Systems* 33 (1), 64–95. doi:10.1016/j.is.2007.07.001.

Rummler, G.A., Brache, A.P. (2012). Improving Performance: How to Manage the White Space on the Organization Chart. 3rd ed., Jossey-Bass, San Francisco.

Scepura, R.C. (2020). The Challenges With Pre-Employment Testing and Potential Hiring Bias. *Nurse Leader* 18 (2), 151–156. doi:10.1016/j.mnl.2019.11.014.

Schönig, S., Rogge-Solti, A., Cabanillas, C., Jablonski, S., Mendling, J. (2016). Efficient and Customisable Declarative Process Mining with SQL, in: Nurcan, S., Soffer, P., Bajec, M., Eder, J. (Eds.), *Advanced Information Systems Engineering. CAiSE 2016*, Springer, Cham. pp. 290–305. doi:10.1007/978-3-319-39696-5_18.

Schulhoff, S., Ilie, M., Balepur, N., Kahadze, K., Liu, A., Si, C., Li, Y., Gupta, A., Han, H., Schulhoff, S., Dulepet, P.S., Vidyadhara, S., Ki, D., Agrawal, S., Pham, C., Kroiz, G., Li, F., Tao, H., Srivastava, A., Costa, H.D., Gupta, S., Rogers, M.L., Goncearenco, I., Sarli, G., Galynker, I., Peskoff, D., Carpuat, M., White, J., Anadkat, S., Hoyle, A., Resnik, P. (2025). The Prompt Report: A Systematic Survey of Prompt Engineering Techniques. arXiv:2406.06608.

Shuster, K., Poff, S., Chen, M., Kiela, D., Weston, J. (2021). Retrieval Augmentation Reduces Hallucination in Conversation, in: Moens, M.F., Huang, X., Specia, L., Yih, S.W.t. (Eds.), *Findings of the Association for Computational Linguistics. EMNLP 2021*, Punta Cana, Dominican Republic. pp. 3784–3803. doi:10.18653/v1/2021.findings-emnlp.320.

Simon, J.P. (2019). Artificial intelligence: scope, players, markets and geography. *Digital Policy, Regulation and Governance* 21 (3), 208–237. doi:10.1108/DPRG-08-2018-0039.

Smalei, U., Kecht, C., Kratsch, W., Röglinger, M. (2026). Process Improvement Copilot: bridging the gap between process inefficiencies and process improvement ideas. *Process Science* 3 (1). doi:10.1007/s44311-025-00028-2.

Sonnenberg, C., vom Brocke, J. (2012a). Evaluation Patterns for Design Science Research Artefacts, in: Helfert, M., Donnellan, B. (Eds.), *Practical Aspects of Design Science. EDSS 2011*, Springer, Berlin, Heidelberg. pp. 71–83. doi:10.1007/978-3-642-33681-2_7.

Sonnenberg, C., vom Brocke, J. (2012b). Evaluations in the Science of the Artificial – Reconsidering the Build-Evaluate Pattern in Design Science Research, in: Peffers, K., Rothenberger, M., Kuechler, B. (Eds.), *Design Science Research in Information Systems. Advances in Theory and Practice. DESRIST 2012*, Springer, Berlin, Heidelberg. pp. 381–397. doi:10.1007/978-3-642-29863-9_28.

Stachl, C., Pargent, F., Hilbert, S., Harari, G.M., Schoedel, R., Vaid, S., Gosling, S.D., Bühner, M. (2020). Personality Research and Assessment in the Era of Machine Learning. *European Journal of Personality* 34 (5), 613–631. doi:10.1002/per.2257.

Tang, J., Liu, Y., Lin, K., Li, L. (2023). Process bottlenecks identification and its root cause analysis using fusion-based clustering and knowledge graph. *Advanced Engineering Informatics* 55, 101862. doi:10.1016/j.aei.2022.101862.

Tett, R.P., Simonet, D.V. (2021). Applicant Faking on Personality Tests: Good or Bad and Why Should We Care? *Personnel Assessment and Decisions* 7 (1), 6–19. doi:10.25035/pad.2021.01.002.

Thought Vector (2017). Customer Support on Twitter. URL: https://www.kaggle.com/thoughtvector/customer-support-on-twitter. (Licensed under CC BY-NC-SA 4.0).

Truong, T.M., Lê, L.S. (2016). Towards a Formal Framework for Business Process Re-Design Based on Data Mining, in: Schmidt, R., Guédria, W., Bider, I., Guerreiro, S. (Eds.), *Enterprise, Business-Process and Information Systems Modeling. BPMDS EMMSAD 2016*, Ljubljana, Slovenia: Springer. pp. 250–265. doi:10.1007/978-3-319-39429-9_16.

Ukpabi, D.C., Aslam, B., Karjaluoto, H. (2019). Chatbot Adoption in Tourism Services: A Conceptual Exploration, in: Ivanov, S., Webster, C. (Eds.), *Robots, Artificial Intelligence, and Service Automation in Travel, Tourism and Hospitality*. Emerald Publishing Limited, pp. 105–121. doi:10.1108/978-1-78756-687-320191006.

van der Aalst, W. (2012). Process Mining: Overview and Opportunities. *ACM Transactions on Management Information Systems* 3 (2). doi:10.1145/2229156.2229157.

van der Aalst, W. (2016). Process Mining: Data Science in Action. 2nd ed., Springer, Berlin, Heidelberg. doi:10.1007/978-3-662-49851-4.

van der Aalst, W., Adriansyah, A., de Medeiros, A.K.A., Arcieri, F., Baier, T., Blickle, T., Bose, J.C., van den Brand, P., Brandtjen, R., Buijs, J., Burattin, A., Carmona, J., Castellanos, M., Claes, J., Cook, J., Costantini, N., Curbera, F., Damiani, E., de Leoni, M., Delias, P., van Dongen, B.F., Dumas, M., Dustdar, S., Fahland, D., Ferreira, D.R., Gaaloul, W., van Geffen, F., Goel, S., Günther, C., Guzzo, A., Harmon, P., ter Hofstede, A., Hoogland, J., Ingvaldsen, J.E., Kato, K., Kuhn, R., Kumar, A., La Rosa, M., Maggi, F., Malerba, D., Mans, R.S., Manuel, A., McCreesh, M., Mello, P., Mendling, J., Montali, M., Motahari-Nezhad, H.R., zur Muehlen, M., Munoz-Gama, J., Pontieri, L., Ribeiro, J., Rozinat, A., Seguel Pérez, H., Seguel Pérez, R., Sepúlveda, M., Sinur, J., Soffer, P., Song, M., Sperduti, A., Stilo, G., Stoel, C., Swenson, K., Talamo, M., Tan, W., Turner, C., Vanthienen, J., Varvaressos, G., Verbeek, E., Verdonk, M., Vigo, R., Wang, J., Weber, B., Weidlich, M., Weijters, T., Wen, L., Westergaard, M., Wynn, M. (2012). Process Mining Manifesto, in: Daniel, F., Barkaoui, K., Dustdar, S. (Eds.), *Business Process Management Workshops. BPM 2011*, Clermont-Ferrand, France: Springer. pp. 169–194. doi:10.1007/978-3-642-28108-2_19.

van der Aalst, W., Weijters, T., Maruster, L. (2004). Workflow mining: discovering process models from event logs. *IEEE Transactions on Knowledge and Data Engineering* 16 (9), 1128–1142. doi:10.1109/TKDE.2004.47.

van der Aalst, W.M.P. (2011). Process Mining: Discovery, Conformance and Enhancement of Business Processes. 1st ed., Springer, Berlin, Heidelberg. doi:10.1007/978-3-642-19345-3.

van der Aalst, W.M.P. (2019). Object-Centric Process Mining: Dealing with Divergence and Convergence in Event Data, in: Ölveczky, P.C., Salaün, G. (Eds.), *Software Engineering and Formal Methods. SEFM 2019*, Springer, Cham. pp. 3–25. doi:10.1007/978-3-030-30446-1_1.

van der Aalst, W.M.P. (2023a). Object-Centric Process Mining: An Introduction, in: Cerone, A. (Ed.), *Formal Methods for an Informal World. ICTAC 2021*, Astana, Kazakhstan: Springer, Cham. pp. 73–105. doi:10.1007/978-3-031-43678-9_3.

van der Aalst, W.M.P. (2023b). Object-Centric Process Mining: Unraveling the Fabric of Real Processes. *Mathematics* 11 (12). doi:10.3390/math11122691.

van der Aalst, W.M.P., Nikolov, A. (2008). Mining E-Mail Messages: Uncovering Interaction Patterns and Processes using E-Mail Logs. *International Journal of Intelligent Information Technologies* 4 (3), 27–45. doi:10.4018/jiit.2008070102.

van Dongen, B. (2017). BPI Challenge 2017. doi:10.4121/uuid:5f3067df-f10b-45da-b98b-86ae4c7a310b.

van Dongen, B. (2019). BPI Challenge 2019. doi:10.4121/uuid:d06aff4b-79f0-45e6-8ec8-e19730c248f1.

van Dongen, B.F., de Medeiros, A.K.A., Verbeek, H.M.W., Weijters, A.J.M.M., van der Aalst, W.M.P. (2005). The ProM Framework: A New Era in Process Mining Tool Support, in: Ciardo, G., Darondeau, P. (Eds.), *Applications and Theory of Petri Nets 2005. ICATPN 2005*, Springer, Berlin, Heidelberg. pp. 444–454. doi:10.1007/11494744_25.

van Dongen, B.F., Mendling, J., van der Aalst, W.M.P. (2006). Structural Patterns for Soundness of Business Process Models, in: *2006 10th IEEE International Enterprise Distributed Object Computing Conference (EDOC'06)*, Hong Kong, China: IEEE. pp. 116–128. doi:10.1109/EDOC.2006.56.

van Dun, C., Moder, L., Kratsch, W., Röglinger, M. (2023). ProcessGAN: Supporting the creation of business process improvement ideas through generative machine learning. *Decision Support Systems* 165, 113880. doi:10.1016/j.dss.2022.113880.

Vanwersch, R.J.B., Shahzad, K., Vanderfeesten, I., Vanhaecht, K., Grefen, P., Pintelon, L., Mendling, J., van Merode, G.G., Reijers, H.A. (2016). A Critical Evaluation and Framework of Business Process Improvement Methods. *Business & Information Systems Engineering* 58 (1), 43–53. doi:10.1007/s12599-015-0417-x.

Varela, J.G., Boccaccini, M.T., Scogin, F., Stump, J., Caputo, A. (2004). Personality Testing in Law Enforcement Employment Settings: A Metaanalytic Review. *Criminal Justice and Behavior* 31 (6), 649–675. doi:10.1177/0093854804268746.

Vasiliev, Y. (2020). Natural Language Processing with Python and spaCy: A Practical Introduction. No Starch Press.

Vaswani, A., Shazeer, N., Parmar, N., Uszukoreit, J., Jones, L., Gomez, A., Kaiser, L., Polosukhin, I. (2017). Attention is all you need, in: Guyon, I., Von Luxburg, U., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R. (Eds.), *Advances in Neural Information Processing Systems 30 (NIPS 2017)*, Long Beach, CA, USA. pp. 6000–6010.

Venable, J., Pries-Heje, J., Baskerville, R. (2016). FEDS: a Framework for Evaluation in Design Science Research. *European Journal of Information Systems* 25 (1), 77–89. doi:10.1057/ejis.2014.36.

Vergidis, K., Tiwari, A., Majeed, B. (2006). Business process improvement using multi-objective optimisation. *BT Technology Journal* 24 (2), 229–235. doi:10.1007/s10550-006-0065-2.

Volk, S., Thöni, C., Ruigrok, W. (2011). Personality, personal values and cooperation preferences in public goods games: A longitudinal study. *Personality and Individual Differences* 50 (6), 810–815. doi:10.1016/j.paid.2011.01.001.

vom Brocke, J., Zelt, S., Schmiedel, T. (2016). On the role of context in business process management. *International Journal of Information Management* 36 (3), 486–495. doi:10.1016/j.ijinfomgt.2015.10.002.

Wastell, D.G., White, P., Kawalek, P. (1994). A methodology for business process redesign: experiences and issues. *The Journal of Strategic Information Systems* 3 (1), 23–40. doi:10.1016/0963-8687(94)90004-3.

Weijters, A.J.M.M., van der Aalst, W.M.P., Alves de Medeiros, A.K. (2006). Process mining with the HeuristicsMiner algorithm .

Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., Davison, J., Shleifer, S., von Platen, P., Ma, C., Jernite, Y., Plu, J., Xu, C., Scao, T.L., Gugger, S., Drame, M., Lhoest, Q., Rush, A.M. (2020). Transformers: State-of-the-Art Natural Language Processing, in: Liu, Q., Schlangen, D. (Eds.), *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, Online. pp. 38–45. doi:10.18653/v1/2020.emnlp-demos.6.

Wynn, M.T., Lebherz, J., van der Aalst, W.M.P., Accorsi, R., Di Ciccio, C., Jayarathna, L., Verbeek, H.M.W. (2022). Rethinking the Input for Process Mining: Insights from the XES Survey and Workshop, in: Munoz-Gama, J., Lu, X. (Eds.), *Process Mining Workshops. ICPM 2021*, Eindhoven, Netherlands: Springer, Cham. pp. 3–16. doi:10.1007/978-3-030-98581-3_1.

Ye, H., Liu, T., Zhang, A., Hua, W., Jia, W. (2024). Cognitive Mirage: A Review of Hallucinations in Large Language Models, in: Zhang, N., Wu, T., Wang, M., Qi, G., Wang, H., Chen, H. (Eds.), *Proceedings of the First International OpenKG Workshop: Large Knowledge-Enhanced Models, co-locacted with The International Joint Conference on Artificial Intelligence. IJCAI 2024*, Jeju Island, South Korea. pp. 14–36.

Yin, W., Hay, J., Roth, D. (2019). Benchmarking Zero-shot Text Classification: Datasets, Evaluation and Entailment Approach, in: Inui, K., Jiang, J., Ng, V., Wan, X. (Eds.), *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, Hong Kong, China. pp. 3914–3923. doi:10.18653/v1/D19-1404.

Zellner, G. (2011). A structured evaluation of business process improvement approaches. *Business Process Management Journal* 17 (2), 203–237. doi:10.1108/14637151111122329.

Zhao, Y., Zhang, R., Li, W., Li, L. (2025). Assessing and Understanding Creativity in Large Language Models. *Machine Intelligence Research* 22 (3), 417–436. doi:10.1007/s11633-025-1546-4.

Zhou, L., Burgoon, J.K., Nunamaker, J.F., Twitchell, D. (2004). Automating Linguistics-Based Cues for Detecting Deception in Text-Based Asynchronous Computer-Mediated Communications. *Group Decision and Negotiation* 13 (1), 81–106. doi:10.1023/B:GRUP.0000011944.62889.6f.

Ziche, C., Apruzzese, G. (2024). LLM4PM: A case study on using Large Language Models for Process Modeling in Enterprise Organizations, in: Di Ciccio, C., Fdhila, W., Agostinelli, S., Amyot, D., Leopold, H., Krčál, M., Malinova Mandelburger, M., Polančič, G., Tomičić-Pupek, K., Gdowska, K., Grisold, T., Sliž, P., Beerepoot, I., Gabryelczyk, R., Plattfaut, R. (Eds.), *Business Process Management: Blockchain, Robotic Process Automation, Central and Eastern European, Educators and Industry Forum. BPM 2024*, Krakow, Poland: Springer. pp. 472–483. doi:10.1007/978-3-031-70445-1_35.

A Appendix

A.1 Index of Research Articles

RA 1: Event Log Construction from Customer Service Conversations Using Natural Language Inference

Kecht, C.; Egger, A.; Kratsch, W.; Röglinger, M. (2021). "Event Log Construction from Customer Service Conversations Using Natural Language Inference". Published in: *3rd International Conference on Process Mining (ICPM)*, Eindhoven, Netherlands. DOI: `https://doi.org/10.1109/ICPM53251.2021.9576869`.
(VHB-24[2]: B, VHB-JQ3[3]: -, Acceptance Rate: 24%)

RA 2: Process Mining between the Lines: Extracting Object-Centric Event Logs from Textual Data

Buss, A.; Kecht, C.; Kratsch, W.; Röglinger, M.; Sadeghianasl, S.; Wynn, M. T. (2026). "Process Mining between the Lines: Extracting Object-Centric Event Logs from Textual Data". Published in: *Information Systems*. DOI: `https://doi.org/10.1016/j.is.2026.102713`.
(VHB-24: B, VHB-JQ3: B, SJR[4]: Q1, IF[5]: 3.4)

An earlier version of RA 2 has been published as follows:

Buss, A.; Kecht, C.; Kratsch, W.; Röglinger, M.; Sadeghianasl, S.; Wynn, M. T. (2025). "From Words to Workflows: Extracting Object-Centric Event Logs from Textual Data". Published in: *Intelligent Information Systems. CAiSE 2025*, Vienna, Austria. Lecture Notes in Business Information Processing, vol 557. DOI: `https://doi.org/10.1007/978-3-031-94590-8_5`.
(VHB-24: C, VHB-JQ3: -)

RA 3: Process Improvement Copilot: Bridging the Gap between Process Inefficiencies and Process Improvement Ideas

Smalei, U.; Kecht, C.; Kratsch, W.; Röglinger, M. (2026). "Process Improvement Copilot: Bridging the Gap between Process Inefficiencies and Process Improvement Ideas". Published in: *Process Science*. DOI: `https://doi.org/10.1007/s44311-025-00028-2`.
(VHB-24: -, VHB-JQ3: -)

[2] VHB-24: VHB Publication Media Rating 2024
[3] VHB-JQ3: VHB-JOURQUAL3
[4] SJR: Scimago Journal & Country Rank 2024
[5] Impact Factor 2024

RA 4: Thinking Outside the Log: Automated Business Process Improvement Using Large Language Models

Bitenc, C.; Dechert, F.; Kecht, C.; Kratsch, W.; Röglinger, M. "Thinking Outside the Log: Automated Business Process Improvement Using Large Language Models". *Working Paper*.

RA 5: Quantifying Chatbots' Ability to Learn Business Processes[6] [7]

Kecht, C.; Egger, A.; Kratsch, W.; Röglinger, M. (2023). "Quantifying Chatbots' Ability to Learn Business Processes". Published in: *Information Systems*. DOI: `https://doi.org/10.1016/j.is.2023.102176`.
(VHB-24: B, VHB-JQ3: B, SJR: Q1, IF: 3.4)

RA 6: Revival of the Cover Letter? Experimental Evidence on the Performance of AI-driven Personality Assessments

Kecht, C.; Kurschilgen, M.; Strobel, M. (2022). "Revival of the Cover Letter? Experimental Evidence on the Performance of AI-driven Personality Assessments". Published in: *Proceedings of the 43rd International Conference on Information Systems (ICIS)*, Copenhagen, Denmark.
(VHB-24: A, VHB-JQ3: A, Acceptance Rate: 26%)

Over the course of my PhD journey, I also contributed to the following RAs. These RAs are not part of this thesis.

Dombetzki, L.; Kecht, C.; Kratsch, W.; Rau, D. (2020). "AMARYLLIS: A User-Centric Information System for Automated Privacy Policy Analysis". Published in: *Proceedings of the 28th European Conference on Information Systems (ECIS) - A Virtual AIS Conference*.

Baumgarte, F.; Dombetzki, L.; Kecht, C.; Wolf, L.; Keller, R. (2021). "AI-based Decision Support for Sustainable Operation of Electric Vehicle Charging Parks". Published in: *Proceedings of the 54th Hawaii International Conference on System Sciences*.

[6] In 2023, my co-authors and I accepted an invitation to present RA 5 in the journal-first track of the *21st International Conference on Business Process Management* in Utrecht, The Netherlands.

[7] In 2024, Andreas Egger, Wolfgang Kratsch, and I were awarded the "Young Researcher Best Paper Award" by the association "Die Wirtschaftsinformatik e.V." for our contributions to RA 5.

A.2 Individual Contribution to the Research Articles

This cumulative thesis comprises six RAs written in collaboration with multiple co-authors. Even though my co-authors' contributions were indispensable to the success of each project, I made a substantial contribution to the respective RAs. In the following paragraphs, I describe my individual contribution to each RA.

In RA 1 (Kecht et al., 2021), entitled "Event Log Construction from Customer Service Conversations Using Natural Language Inference" and written by a team of four co-authors, I initiated the research project, designed and implemented the NLI-based pipeline, prepared the datasets, and evaluated the developed approach. I wrote the original manuscript, coordinated the responses to the reviewers, and served as the corresponding author throughout the peer review process of the conference. As agreed by the team, one co-author is listed as a subordinate author, while the other two co-authors and I contributed equally to the project.

In RA 2 (Buss et al., 2026), entitled "Process Mining between the Lines: Extracting Object-Centric Event Logs from Textual Data" and written by a team of six co-authors, I co-initiated the project and contributed substantially to implementing and evaluating the generative collector and refiner. Furthermore, I had a key role in drafting major parts of the manuscript and overseeing the project. As a team, we agreed that all authors contributed equally to the project.

In RA 3 (Smalei et al., 2026), entitled "Process Improvement Copilot: Bridging the Gap between Process Inefficiencies and Process Improvement Ideas" and written by a team of four co-authors, I contributed to the conceptualization and methodology. Furthermore, I was deeply involved in reviewing and editing the manuscript, and played a central role in revising the article in response to the reviewers' feedback. As a team, we agreed that all authors contributed equally to the project.

In RA 4, entitled "Thinking Outside the Log: Automated Business Process Improvement Using Large Language Models" and written by a team of five co-authors, I contributed to the conceptualization and methodology, supported the implementation of the artifact, and had a key role in reviewing and editing the manuscript prior to submission. As a team, we agreed that all authors contributed equally to the project.

In RA 5 (Kecht et al., 2023), entitled "Quantifying Chatbots' Ability to Learn Business Processes" and written by a team of four co-authors, I originated the research idea, developed and implemented the approach, and wrote the original draft of the manuscript. I coordinated the revisions during the peer review process of the journal and finalized the paper for publication. As agreed by the team, one co-author is listed as a subordinate author, while the other two co-authors and I contributed equally to the project.

In RA 6 (Kecht et al., 2022), entitled “Revival of the Cover Letter? Experimental Evidence on the Performance of AI-driven Personality Assessments” and written by a team of three co-authors, I served as the leading author with a key role across all stages of the project. In particular, I contributed to the research design, implemented and evaluated the machine learning approach, wrote large parts of the manuscript, managed the revision during the peer review process of the conference, and finally presented the article at the 43rd International Conference on Information Systems (ICIS) in Copenhagen, Denmark.

A.3 Research Article 1: Event Log Construction from Customer Service Conversations Using Natural Language Inference

Authors:
Christoph Kecht; Andreas Egger; Wolfgang Kratsch; Maximilian Röglinger

Published in:
3rd International Conference on Process Mining (ICPM), Eindhoven, Netherlands (2021)

Abstract:
A fundamental requirement for the successful application of process mining are event logs of high data quality that can be constructed from structured data stored in organizations' core information systems. However, a substantial amount of data is processed outside these core systems, particularly in organizations doing consumer business with many customer interactions per day, which generate high amounts of unstructured text data. Although Natural Language Processing (NLP) and machine learning enable the exploitation of text data, these approaches remain challenging due to the required high amount of labeled training data. Recent advances in NLP mitigate this issue by providing pre-trained and ready-to-use language models for various tasks such as Natural Language Inference (NLI). In this paper, we develop an approach that utilizes NLI to derive topics and process activities from customer service conversations and that represents them in a standardized XES event log. To this end, we compute the probability that a sentence describing the topic or the process activity can be inferred from the customer's inquiry or the agent's response using NLI. We evaluate our approach utilizing an existing corpus of more than 500,000 customer service conversations of three companies on Twitter. The results show that NLI helps construct event logs of high accuracy for process mining purposes, as our successful application of three different process discovery algorithms confirms.

Keywords:
Process Mining, Event Log Construction, Machine Learning, Natural Language Processing

A.4 Research Article 2: Process Mining between the Lines: Extracting Object-Centric Event Logs from Textual Data

Authors:
Alina Buss; Christoph Kecht; Wolfgang Kratsch; Maximilian Röglinger; Sareh Sadeghianasl; Moe T. Wynn

Published in:
Information Systems (2026)

Abstract:
Organizations generate vast amounts of unstructured textual data – a valuable source of information that frequently remains underutilized for process mining. However, textual descriptions often record exceptions and manual activities absent from structured data, and therefore, enable a better understanding of deviations from the expected business process behavior. Importantly, unstructured sources typically retain the object-centric characteristics of real-world processes – information that gets flattened or lost in case-centric event logs. Yet, existing approaches primarily target structured data sources or produce case-centric event logs. To address this gap, we present an automated approach to derive object-centric event logs directly from unstructured textual descriptions. The approach comprises two subcomponents: a *collector* that identifies events and objects (including their attributes and relationships), and a *refiner* that consolidates and cleans the extracted information. We instantiate each subcomponent in heuristic and generative implementations and create four pairwise combinations of collector and refiner instances to assess the effectiveness of heuristic natural language processing and generative artificial intelligence techniques. We compare these variants quantitatively and qualitatively in a controlled, artificial setting based on synthesized texts and demonstrate the practical utility on two naturally occurring corpora (fire status updates and a legal judgment). Our results show that the configurations with a generative collector achieve the highest extraction quality. In particular, the fully generative variant produces coherent and standardized event and object labels. Overall, this study fills a notable research gap by enabling the incorporation of textual information into process mining applications.

Keywords:
Process Mining, Object-Centric Event Logs, Natural Language Processing, Large Language Models, Generative Artificial Intelligence

A.5 Research Article 3: Process Improvement Copilot: Bridging the Gap between Process Inefficiencies and Process Improvement Ideas

Authors:
Uladzimir Smalei; Christoph Kecht; Wolfgang Kratsch; Maximilian Röglinger

Published in:
Process Science (2026)

Abstract:
Business process improvement (BPI) is a crucial value-adding stage of business process management, as it introduces process changes to eliminate flaws and enhance performance. However, the inherent demands of BPI on domain knowledge, process expertise, time, and creativity in conjunction with a scarcity of adequate computational support, hinder organizations from fully leveraging BPI. Recognizing this gap, recent research calls for all types of contributions to process improvement and innovation systems (PIISs), from design knowledge to software artifacts. Leveraging the latest developments in generative artificial intelligence, increased availability of process execution data, and extensive collections of BPI knowledge, we propose a new technical approach to supporting the generation of process improvement ideas in BPI initiatives. To this end, we develop the Process Improvement Copilot – a retrieval-augmented generation (RAG)-enhanced PIIS that generates context-specific process improvement ideas and provides related justification, thereby facilitating their further evaluation and implementation. This research contributes a novel technical approach to automated BPI by exploring a RAG-based use case, designing a corresponding system architecture, developing a software prototype to demonstrate its technical feasibility, and evaluating the Process Improvement Copilot's usefulness in a naturalistic workshop setting.

Keywords:
Business Process Management, Business Process Improvement, Process Mining, Generative Artificial Intelligence, Retrieval-Augmented Generation

A.6 Research Article 4: Thinking Outside the Log: Automated Business Process Improvement Using Large Language Models

Authors:
Camille Bitenc; Franziska Dechert; Christoph Kecht; Wolfgang Kratsch; Maximilian Röglinger

Extended Abstract:

Organizations need to constantly improve their business processes to succeed in competitive environments (Vergidis et al., 2006). Business Process Improvement (BPI) is inherently creative (Figl and Recker, 2016; van Dun et al., 2023), requires human ingenuity (Dumas et al., 2018), and involves domain experts to this day (Mustansir et al., 2022). These characteristics make resource availability, particularly in terms of time and labor (Beerepoot et al., 2023), an important contextual factor for BPI since traditional creativity techniques such as brainstorming (Kettinger et al., 1997) are very time- and resource-intensive.

In response to the resource-intensiveness of BPI projects, extant research explores approaches to (semi-)automate BPI. Many of those approaches focus on the control flow, which reveals the arrangement of activities in a process model and hence represents the backbone of every process model (van der Aalst et al., 2012). The control flow is relevant to BPI as it provides a starting point for BPI initiatives in the form of an as-is process model (Malinova et al., 2022), which is needed for incremental improvement (Davenport, 1993). However, extant approaches fall short in at least one essential BPI capability, such as process evaluation (Malinova et al., 2022), inclusion of process context factors (Moder et al., 2025), inclusion of (multiple) improvement objectives (Vergidis et al., 2006), or creativity (Gross et al., 2021). Creative techniques lead to a larger and more diverse solution space of redesigns (Groß et al., 2024), positioning creativity as an essential element for BPI. Yet, the solution spaces of most existing approaches are restricted.

Synthesizing computational creativity (Zhao et al., 2025) with strong reasoning capabilities (Kojima et al., 2022) and hence being able to search for BPI ideas beyond restrictive boundaries, makes Large Language Models (LLMs) a prime candidate for automating BPI. LLMs have already been employed in several business process management activities, such as explaining process models (Fahland et al., 2025), supporting the creation of process models (Ziche and Apruzzese, 2024) or even modeling processes by themselves from textual descriptions (Kourani et al., 2024; Köpke and Safan, 2025).

To investigate how LLMs can support BPI initiatives by suggesting improved business process models and how the suggested improvements can be quantified, we follow the Design Science Research (DSR) paradigm (Gregor and Hevner, 2013) and structure our work along the DSR methodology of Peffers et al. (2007). Through this process, we design, instantiate, and evaluate the Automated Business Process Optimizer (ABuPrOpt), which is designed to fulfill three Design Objectives (DOs) derived from extant literature to enable the targeted generation and quantitative evaluation of improved (i. e. better performing), sound (i. e. syntactically correct, as elaborated by van Dongen et al. 2006), and feasible (i. e. semantically correct) process models. ABuPrOpt im-

proves process models based on the control-flow perspective (van der Aalst, 2016) and comprises three core components: the improvement generator, the improvement evaluator, and the User Interface (UI). The improvement generator leverages an LLM to generate improved, sound, and feasible process models while incorporating standard and custom improvement objectives. To compare the generated process model to the original model, the improvement evaluator assesses the respective time and cost performance through simulation. The UI serves as a convenient layer for both process analysts and domain experts to operate ABuPrOpt.

We evaluate ABuPrOpt in three distinct evaluation episodes based on the Framework for Evaluation in Design Science (FEDS) by Venable et al. (2016). Since the major design risk in our work is technically oriented (i. e. technical feasibility of a design based on LLMs and simulation), we employ the Technical Risk & Efficacy strategy of FEDS, progressively advancing from formative to summative evaluation episodes. First, in an artificial, formative setting, we perform a literature review to identify the underlying research gap of automating BPI and justifying LLMs as a possible solution. Second, in an artificial and summative setting, we perform a competing artifact analysis measuring ABuPrOpt's and extant approaches' alignment with our DOs. Third, in a naturalistic and summative setting, we demonstrate ABuPrOpt's functionality to improve and assess process models derived from five publicly available data sets, thus evaluating the efficacy of the software prototype and its fulfillment of the DOs in a practical environment.

We find that ABuPrOpt is able to develop and quantitatively simulate improved process models in a targeted manner while considering inferred dependencies between activities. Our work adds to existing literature through an approach that automates core capabilities of BPI, having a solution space expanding beyond boundaries imposed by process input data or externally defined instructional frameworks like patterns or rules. Hence, ABuPrOpt is able to capture the original process model semantically and therefore provides feasible process models by design. In addition, ABuPrOpt enables the inclusion of standard and custom improvement objectives, thus maximizing the degrees of freedom for finding tailored BPI ideas. The improvement evaluator consolidates the contribution through the capability to handle entirely novel activities as well as variants that are not present in the event log of the original model.

Keywords:
Business Process Management, Business Process Improvement, Process Mining, Generative AI, Large Language Models, Business Process Simulation

References:

Beerepoot, I., Di Ciccio, C., Reijers, H.A., Rinderle-Ma, S., Bandara, W., Burattin, A., Calvanese, D., Chen, T., Cohen, I., Depaire, B., Di Federico, G., Dumas, M., van Dun, C., Fehrer, T., Fischer, D.A., Gal, A., Indulska, M., Isahagian, V., Klinkmüller, C., Kratsch, W., Leopold, H., Van Looy, A., Lopez, H., Lukumbuzya, S., Mendling, J., Meyers, L., Moder, L., Montali, M., Muthusamy, V., Reichert, M., Rizk, Y., Rosemann, M., Röglinger, M., Sadiq, S., Seiger, R., Slaats, T., Simkus, M., Someh, I.A., Weber, B., Weber, I., Weske, M., Zerbato, F. (2023). The biggest business process management problems to solve before we die. *Computers in Industry* 146, 103837. doi:10.1016/j.compind.2022.103837.

Davenport, T.H. (1993). Process innovation: reengineering work through information technology. Harvard Business School Press, Boston, MA, USA.

Dumas, M., La Rosa, M., Mendling, J., Reijers, H.A. (2018). Process Redesign, in: *Fundamentals of Business Process Management*. Springer, Berlin, Heidelberg, pp. 297–339. doi:10.1007/978-3-662-56509-4_8.

Fahland, D., Fournier, F., Limonad, L., Skarbovsky, I., Swevels, A.J.E. (2025). How well can a large language model explain business processes as perceived by users? *Data & Knowledge Engineering* 157, 102416. doi:10.1016/j.datak.2025.102416.

Figl, K., Recker, J. (2016). Process innovation as creative problem solving: An experimental study of textual descriptions and diagrams. *Information & Management* 53 (6), 767–786. doi:10.1016/j.im.2016.02.008.

Gregor, S., Hevner, A.R. (2013). Positioning and presenting design science research for maximum impact. *MIS Quarterly* 37 (2), 337–355. doi:10.25300/MISQ/2013/37.2.01.

Groß, S., Grisold, T., Mendling, J., Haase, J. (2024). Idea generation in exploitative and explorative business process redesign techniques. *Information Systems and e-Business Management* 22 (3), 527–555. doi:10.1007/s10257-024-00684-0.

Gross, S., Stelzl, K., Grisold, T., Mendling, J., Röglinger, M., vom Brocke, J. (2021). The Business Process Design Space for exploring process redesign alternatives. *Business Process Management Journal* 27 (8), 25–56. doi:10.1108/BPMJ-03-2020-0116.

Kettinger, W.J., Teng, J.T.C., Guha, S. (1997). Business Process Change: A Study of Methodologies, Techniques, and Tools. *MIS Quarterly* 21 (1), 55–80. doi:10.2307/249742.

Kojima, T., Gu, S.S., Reid, M., Matsuo, Y., Iwasawa, Y. (2022). Large Language Models are Zero-Shot Reasoners, in: Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., Oh, A. (Eds.), *Advances in Neural Information Processing Systems 35 (NeurIPS 2022)*, New Orleans, LA, USA. pp. 22199–22213.

Kourani, H., Berti, A., Schuster, D., van der Aalst, W.M.P. (2024). Process Modeling with Large Language Models, in: van der Aa, H., Bork, D., Schmidt, R., Sturm, A. (Eds.), *Enterprise, Business-Process and Information Systems Modeling. BPMDS EMMSAD 2024*, Limassol, Cyprus: Springer. pp. 229–244. doi:10.1007/978-3-031-61007-3_18.

Köpke, J., Safan, A. (2025). Efficient LLM-Based Conversational Process Modeling, in: Gdowska, K., Gómez-López, M.T., Rehse, J.R. (Eds.), *Business Process Management Workshops. BPM 2024*, Krakow, Poland: Springer. pp. 259–270. doi:10.1007/978-3-031-78666-2_20.

Malinova, M., Gross, S., Mendling, J. (2022). A study into the contingencies of process improvement methods. *Information Systems* 104, 101880. doi:10.1016/j.is.2021.101880.

Moder, L., Fehrer, T., Röglinger, M. (2025). Design principles for process improvement and innovation systems. *Business Process Management Journal* doi:10.1108/BPMJ-10-2024-0975.

Mustansir, A., Shahzad, K., Malik, M.K. (2022). Towards automatic business process redesign: an NLP based approach to extract redesign suggestions. *Automated Software Engineering* 29 (1), 12. doi:10.1007/s10515-021-00316-8.

Peffers, K., Tuunanen, T., Rothenberger, M.A., Chatterjee, S. (2007). A Design Science Research Methodology for Information Systems Research. *Journal of Management Information Systems* 24 (3), 45–77. doi:10.2753/MIS0742-1222240302.

van der Aalst, W. (2016). Process Mining: Data Science in Action. 2nd ed., Springer, Berlin, Heidelberg. doi:10.1007/978-3-662-49851-4.

van der Aalst, W., Adriansyah, A., de Medeiros, A.K.A., Arcieri, F., Baier, T., Blickle, T., Bose, J.C., van den Brand, P., Brandtjen, R., Buijs, J., Burattin, A., Carmona, J., Castellanos, M., Claes, J., Cook, J., Costantini, N., Curbera, F., Damiani, E., de Leoni, M., Delias, P., van Dongen, B.F., Dumas, M., Dustdar, S., Fahland, D., Ferreira, D.R., Gaaloul, W., van Geffen, F., Goel, S., Günther, C., Guzzo, A., Harmon, P., ter Hofstede, A., Hoogland, J., Ingvaldsen, J.E., Kato, K., Kuhn, R., Kumar, A., La Rosa, M., Maggi, F., Malerba, D., Mans, R.S., Manuel, A., McCreesh, M., Mello, P., Mendling, J., Montali, M., Motahari-Nezhad, H.R., zur Muehlen, M., Munoz-Gama, J., Pontieri, L., Ribeiro, J., Rozinat, A., Seguel Pérez, H., Seguel Pérez, R., Sepúlveda, M., Sinur, J., Soffer, P., Song, M., Sperduti, A., Stilo, G., Stoel, C., Swenson, K., Talamo, M., Tan, W., Turner, C., Vanthienen, J., Varvaressos, G., Verbeek, E., Verdonk, M., Vigo, R., Wang, J., Weber, B., Weidlich, M., Weijters, T., Wen, L., Westergaard, M., Wynn, M. (2012). Process Mining Manifesto, in: Daniel, F., Barkaoui, K., Dustdar, S. (Eds.), *Business Process Management Workshops. BPM 2011*, Clermont-Ferrand, France: Springer. pp. 169–194. doi:10.1007/978-3-642-28108-2_19.

van Dongen, B.F., Mendling, J., van der Aalst, W.M.P. (2006). Structural Patterns for Soundness of Business Process Models, in: *2006 10th IEEE International Enterprise Distributed Object Computing Conference (EDOC'06)*, Hong Kong, China: IEEE. pp. 116–128. doi:10.1109/EDOC.2006.56.

van Dun, C., Moder, L., Kratsch, W., Röglinger, M. (2023). ProcessGAN: Supporting the creation of business process improvement ideas through generative machine learning. *Decision Support Systems* 165, 113880. doi:10.1016/j.dss.2022.113880.

Venable, J., Pries-Heje, J., Baskerville, R. (2016). FEDS: a Framework for Evaluation in Design Science Research. *European Journal of Information Systems* 25 (1), 77–89. doi:10.1057/ejis.2014.36.

Vergidis, K., Tiwari, A., Majeed, B. (2006). Business process improvement using multi-objective optimisation. *BT Technology Journal* 24 (2), 229–235. doi:10.1007/s10550-006-0065-2.

Zhao, Y., Zhang, R., Li, W., Li, L. (2025). Assessing and Understanding Creativity in Large Language Models. *Machine Intelligence Research* 22 (3), 417–436. doi:10.1007/s11633-025-1546-4.

Ziche, C., Apruzzese, G. (2024). LLM4PM: A case study on using Large Language Models for Process Modeling in Enterprise Organizations, in: Di Ciccio, C., Fdhila, W., Agostinelli, S., Amyot, D., Leopold, H., Krčál, M., Malinova Mandelburger, M., Polančič, G., Tomičić-Pupek, K., Gdowska, K., Grisold, T., Sliż, P., Beerepoot, I., Gabryelczyk, R., Plattfaut, R. (Eds.), *Business Process Management: Blockchain, Robotic Process Automation, Central and Eastern European, Educators and Industry Forum. BPM 2024*, Krakow, Poland: Springer. pp. 472–483. doi:10.1007/978-3-031-70445-1_35.

A.7 Research Article 5: Quantifying Chatbots' Ability to Learn Business Processes

Authors:
Christoph Kecht; Andreas Egger; Wolfgang Kratsch; Maximilian Röglinger

Published in:
Information Systems (2023)

Abstract:
Chatbots enable organizations in the business-to-customer domain to respond to repetitive requests efficiently. Extant approaches in Natural Language Processing (NLP) already address the essential requirement of understanding user input and synthesizing a response as close as possible to a response a human interlocutor would give. However, we argue that the organizational adoption of chatbots further depends on the underlying model's capability to learn and comply with organizations' business processes, for example, authenticating a customer before providing sensitive details. To address this issue, we develop an approach that quantifies chatbots' ability to learn business processes using standardized process mining metrics. We demonstrate our approach by training chatbots on a dataset of more than 500,000 customer service conversations from three companies on Twitter and show how our approach supports the quantification of a chatbot's overall ability to learn business processes from the training data. Furthermore, we quantify a chatbot's ability to learn a particular variant of the underlying process and we show how to compare the chatbot's executed steps against a given normative process model. Our approach that seamlessly integrates with existing approaches to evaluate NLP-based chatbots mitigates the current hurdles that practitioners face and, therefore, strives to foster the adoption of chatbots in practice.

Keywords:
Chatbots, Process Mining, Natural Language Processing, Conformance Checking

A.8 Research Article 6: Revival of the Cover Letter? Experimental Evidence on the Performance of AI-driven Personality Assessments

Authors:
Christoph Kecht; Michael Kurschilgen; Magnus Strobel

Published in:
Proceedings of the 43rd International Conference on Information Systems (ICIS), Copenhagen, Denmark (2022)

Abstract:
Organizations have long been trying to assess job applicants' personality using self-reported psychometric tests, such as the Big Five personality test. However, these tests are not robust against incentives to pretend having certain desirable traits, for example, the disposition for being a good team player. We test whether machine learning classifiers trained on written self-descriptions, such as cover letters, predict people's true cooperativeness better than psychometric tests. Based on data from a controlled online experiment with 400 participants, we find that – when people have incentives to fake their personality – linguistic classifiers based on self-descriptions significantly outperform psychometric classifiers based on the Big Five. Moreover, we find that a fine-tuned, pre-trained natural language model can detect incentives to fake in people's self-descriptions. While further research is needed to achieve tamper-proof models, our findings illustrate the potential of automated personality tests based on job applicants' cover letters.

Keywords:
Personality Assessment, Cooperativeness, Big Five, Linguistic Inquiry and Word Count, Machine Learning, Natural Language Processing