Smalei *et al. Process Science* (2026) 3:1
https://doi.org/10.1007/s44311-025-00028-2

Process Science

RESEARCH Open Access

Process Improvement Copilot: bridging the gap between process inefficiencies and process improvement ideas

Uladzimir Smalei[1*], Christoph Kecht[2,3,4], Wolfgang Kratsch[2,3,5] and Maximilian Röglinger[2,3,4]

*Correspondence:
Uladzimir Smalei
uladzimir.smalei@tum.de
[1]TUM School of Management, Technical University of Munich, Munich, Germany
[2]FIM Research Center for Information Management, Augsburg, Germany
[3]Branch Business & Information Systems Engineering of the Fraunhofer FIT, Augsburg, Germany
[4]University of Bayreuth, Bayreuth, Germany
[5]Augsburg Technical University of Applied Sciences, Augsburg, Germany

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

Introduction

To succeed in competitive environments, organizations need to constantly improve their business processes (Huang et al. 2015; Kerpedzhiev et al. 2021). Business processes constitute the core of every organization, as they define the activities along which the value creation is coordinated (Becker et al. 2012; Buhl et al. 2011; Dumas et al. 2018; Willaert et al. 2007). Efficient and effective business processes are a prerequisite for achieving key performance indicators (KPIs), such as profitability or customer satisfaction (Groß et al. 2024; Hammer 2015; Huang et al. 2015; Vanwersch et al. 2016). Hence, an increasing number of organizations are tending towards incorporating a process mindset and starting business process improvement (BPI) initiatives (Zellner 2011).

However, BPI initiatives are often time-consuming, require specialized expertise, demand significant creativity, and are inherently complex (Gross et al. 2019; Huang et al. 2015; Limam Mansar et al. 2009). 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). Such manual approaches consume scarce resources and frequently become a bottleneck (Bader et al. 2024; Beerepoot et al. 2019). Limited expert capacity leads to growing improvement backlogs, resulting in unpleasant customer experiences (Kreuzer et al. 2020) and ultimately diminished competitive performance (Huang et al. 2015).

Technological advancements in problem-solving and decision-making systems (Feuerriegel et al. 2024) expanded the range of technical means that can be applied throughout the business process management (BPM) lifecycle. Process mining (PM) – introduced in the early 2000s with a strong focus on the analysis stage – provides data science tools for analyzing process execution data (van der Aalst 2022). In contrast, the BPI stage remains comparatively under-supported due to its inherent complexity (Beerepoot et al. 2023; Röglinger et al. 2021), resulting in a lack of (semi-)automated and scalable process improvement and innovation systems (PIISs) required in real-world scenarios (Ackermann et al. 2024; Fehrer et al. 2025; Khan et al. 2023; Stein Dani et al. 2024).

The rise of generative artificial intelligence (GenAI), with its proven potential to perform creative tasks (Hofmann et al. 2021), addresses a critical and previously mostly unfulfilled creativity requirement towards PIISs (Röglinger et al. 2021). Large language models (LLMs) are the most recent development in the field of GenAI and have been successfully employed to generate and refine process models (Kourani et al. 2024). Although LLMs are strong in generating coherent and contextually rich sequential data like text and handling long-range dependencies (Vaswani et al. 2017), they exhibit two key weaknesses: the generation of inaccurate information ("hallucinations") and a lack of control over the generated output and its origin, making them unsuitable for the generation of contextually relevant BPI ideas in their pure form. To overcome this limitation, retrieval-augmented generation (RAG) enables LLMs to leverage a non-parametric memory of knowledge chunks (Lewis et al. 2020). Motivated by these developments, we strive to answer the research question: *How can we design and develop a RAG-enhanced LLM-based PIIS that supports the generation of process improvement ideas in BPI initiatives?*

We address this research question in the spirit of the design science research (DSR) paradigm (Peffers et al. 2007) and derive design objectives (DOs) that our approach needs to incorporate. Based on the identified DOs, we develop the Process Improvement Copilot by designing a corresponding system architecture and implementing a software prototype iteratively refined based on expert interviews. The proposed Process Improvement Copilot is a RAG-enhanced LLM-based PIIS that supports the generation of process improvement ideas for previously identified process inefficiencies. It 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. Furthermore, it facilitates follow-up actions by justifying each idea and showing the respective pieces of knowledge used to generate each idea.

To illustrate the developed functionalities, we apply the software prototype to an academic dataset of a purchase-to-pay process. We evaluate the DOs through 13 semi-structured interviews with 16 experts from academia and industry, assess the novelty of the Process Improvement Copilot through a competing artifact analysis, and evaluate the system architecture and an initial version of the software prototype with the same panel of experts. Finally, we conduct a process improvement workshop at a multinational technology conglomerate. The majority of the interview experts and workshop participants find the Process Improvement Copilot useful and easy-to-use and agree that it reduces the time and BPI expertise required to generate process improvement ideas. By incorporating existing BPI knowledge in a RAG-based architecture, we contribute a novel approach to automated BPI. Our study facilitates further research on the intersection of GenAI and BPI by publicly sharing the software code.

The remainder of this paper is organized as follows. In Section "Background and related work", we present a literature review focusing on the BPI stage of the BPM lifecycle and relevant artificial intelligence (AI) research. Section "Objectives and requirements definition" derives the DOs that our novel PIIS needs to incorporate. In Section "Design specification", we develop the design specification of our artifact, including the system architecture and the implementation of the software prototype, and illustrate the prototype's functionality using an academic dataset. Section "Demonstration and evaluation" reports the comprehensive evaluation of the Process Improvement Copilot. We conclude the paper in Section "Conclusion" with the contribution, implications for research and practice, and limitations and future research directions.

Background and related work

Business process improvement

BPM is a research field that studies process phenomena with a focus on business processes (Hammer 2015; Reijers 2021). Extant research proposes several BPM lifecycle models, including for example, Dumas et al. (2018), Rosemann and vom Brocke (2015), and van der Aalst (2013). According to Rosemann and vom Brocke (2015), the BPM lifecycle consists of process design and modeling, implementation and execution, monitoring and control, improvement and innovation, and project and program management stages. Regardless of the used BPM lifecycle model, researchers and practitioners usually agree that the process improvement and innovation (PII) stage is the most value-adding (Gross et al. 2021; Kerpedzhiev et al. 2021; Zellner 2011). Following Röglinger et al. (2021), we consider a continuum of the scope of process (re-)design, from incremental (referred to as improvement) to radical (referred to as innovation). The former describes regular, exploitative initiatives with the goal of improving some performance metrics through (minor) enhancements (Malinova et al. 2022; Zellner 2013). The latter, in contrast, describes occasional, explorative initiatives with the goal of developing new ways of value creation in an organization (Beerepoot et al. 2023; Gross et al. 2021; Malinova et al. 2022). The focus of this research is on incremental BPI.

Considering the importance of the PII stage, many researchers contributed to its exploration over the last decades. Kettinger et al. (1997) identified 25 business process reengineering (a closely related and overlapping term for PII) methodologies, which they used to build a composite stage-activity PII framework. This framework consists of the following six stages: *envision*, *initiate*, *diagnose*, *redesign*, *reconstruct*, and *evaluate.* Each stage consists of two to five activities, resulting in a total of 21 activities (Kettinger et al. 1997). Clearly demonstrating the importance of the topic, Malinova et al. (2022) conducted further research into the contingencies of process improvement methods and extended the framework by Kettinger et al. (1997) based on 90 new process improvement methods. This resulted in a significant increase in the number of activities from 21 to 264 (organized in 41 clusters for the six initially identified stages).

For a long time, PII has been a manual stage of the BPM lifecycle (Abbasi et al. 2024; Beerepoot et al. 2023). Some reasons for this are the high levels of process expertise, domain knowledge, and creativity required to conduct PII initiatives (Gross et al. 2019). Moreover, the intricate structure of the PII stage, as shown with the frameworks by Kettinger et al. (1997) and Malinova et al. (2022), requires the involvement of multiple stakeholders from all organizational levels and a nuanced cross-functional understanding of the organizational dynamics (Zuhaira and Ahmad 2021).

Various manual methods and techniques support individual stages of BPI initiatives. For example, the creativity required to develop BPI ideas for the specific process can be enhanced by exploring best practices collected from successful BPI projects (Reijers and Liman Mansar 2005). These best practices are called BPI patterns (also referred to as redesign heuristics or redesign 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). The development of improvement ideas can also be approached from the opposite perspective, namely with anti-patterns. As defined by Koschmider et al. (2019), BPI anti-patterns (also referred to as process weaknesses, weakness patterns, or process flaws) refer to ineffective practices, errors, and violations that should be avoided when improving business processes.

Combining extensive research on BPI patterns and anti-patterns, some researchers proposed ways of logically integrating these two types of practices (Groß et al. 2020; Lashkevich et al. 2023). Both approaches facilitate the transition from PM insights about process inefficiencies to process improvement ideas. Process inefficiencies occur if a business process meets its operational goals but is wasteful in terms of utilized resources (Wastell et al. 1994). However, the matching between process inefficiencies and process improvement ideas is done without consideration of the context and, to some extent, subjectively. Moreover, neither the list of process inefficiencies nor the list of the best practices is exhaustive, and they must be manually re-matched in case of new developments on either side.

A combination of multiple recent technological trends (advancements in imitating human reasoning and decision-making processes by AI systems, increased availability of process data, and advancements in PM techniques) allows researchers to challenge the status quo in the field of computational support for PII and develop tools that "provide (semi-)automated support for the generation of improved business processes," called in research PIISs (Rosemann and vom Brocke 2015, p. 117). Multiple researchers, for example, Ackermann et al. (2024), Beerepoot et al. (2023), Khan et al. (2023), Park and van der Aalst (2022), and Röglinger et al. (2021), encouraged the research community to contribute to the development of PIISs. Moder et al. (2025) present 14 design principles (DPs) covering the current lack of prescriptive design knowledge for PIISs. To describe and analyze PIISs, Fehrer et al. (2025) propose a taxonomy that classifies PIISs based on four layers (*scope*, *input*, *throughput*, and *output*) and eight dimensions. This taxonomy contributes to a better understanding of PIISs and outlines future research directions by clearly distinguishing different design options.

When developing PIISs, researchers are coming from different directions. Fehrer et al. (2022), Fehrer et al. (2024), Niedermann and Schwarz (2011), and Souza et al. (2017) select BPI patterns as a starting point. Although these PIISs incorporate BPI knowledge in the form of BPI patterns, one common limitation of such approaches is that their scalability to new BPI patterns is limited (can only be done manually), which hinders the use of these PIISs in real-world situations with a constantly increasing number of BPI best practices. Another group of PIISs is inspired by anti-patterns (Bergener et al. 2015; Schuh et al. 2021). While these two approaches foresee the ability to support users in identifying process weaknesses, they (similarly to the above-mentioned BPI-pattern-driven PIISs) demonstrate limited scalability to new anti-patterns.

Some authors emphasize the creative nature of the BPI process and try to develop artifacts that incentivize creative solutions. For example, Afflerbach et al. (2017) address the gap of computational support in BPI by developing an evolutionary algorithms application that mimics the BPM lifecycle. Alternatively, van Dun et al. (2023) propose a novel architecture based on generative adversarial network (GANs), an approach similar to evolutionary algorithms inspired by evolutionary biology. A potential limitation of these two approaches is that they do not incorporate existing collections of BPI knowledge.

Recent advancements in GenAI have also attracted the attention of researchers developing PIISs. The architecture proposed by Beheshti et al. (2023) leverages a large dataset of business process data (such as event logs, process models, and BPI best practices) to train a generative pre-trained transformer model called ProcessGPT. Harl et al. (2024) propose a novel method with an "offline phase" where a generative machine learning model is trained on historical process data to predict the as-is process model and an "online phase" where the model predicts necessary changes in real-time, which are then ranked and incorporated into an updated to-be process model. Both projects are currently in progress, and a comprehensive evaluation can only be conducted upon their completion.

Artificial intelligence in business process improvement

AI is usually defined as "the ability of machines to perform cognitive functions that we associate with human minds" (Rai et al. 2019, p. 1). The vast majority of current AI applications fall under the category of artificial narrow intelligence, designed and trained for specific areas or tasks (Kaplan and Haenlein 2019).

One possible area for the application of artificial narrow intelligence is the generation of new content. Hofmann et al. (2021) describe GenAI systems as a class of AI systems that show an ability to generate previously unseen but plausible content, e.g., molecular structures (Chen et al. 2020) or music (Chen et al. 2019). Because of its creative nature, the generation of process improvement ideas in a BPI process is a promising candidate for implementing GenAI. For example, van Dun et al. (2023) explore this with GANs.

However, the landscape of other GenAI architectures already explored in the field of BPI is limited. The application of certain GenAI architectures in BPI might be hindered by specific limitations. For example, traditional LLMs demonstrate two weaknesses restraining them from being used in BPI. First, LLMs can sometimes generate factually incorrect content, referred to as hallucinations (Huang et al. 2025; Maynez et al. 2020; Shuster et al. 2021; Ye et al. 2024). Second, controlling the format and source of the output provided by LLMs is challenging, making it difficult to ensure that the generated content meets specific requirements or constraints (Agrawal et al. 2024). Solving these two weaknesses of LLM-based systems is a prerequisite for developing a valuable LLM-based PIIS.

Lewis et al. (2020) propose the RAG approach and discuss its potential to address the weaknesses of LLMs. According to Lewis et al. (2020), an LLM's ability to precisely manipulate knowledge is limited, as training knowledge is transformed into parameters of the LLM and cannot be used explicitly. However, Lewis et al. (2020) recognize an alternative to parametric knowledge and propose that relevant knowledge might be provided to the LLM non-parametrically. Multiple research groups show that RAG reduces hallucination (Shuster et al. 2021) and improves the factuality and specificity of the response in comparison to LLMs (Lewis et al. 2020). RAG falls between standard LLMs and fine-tuned LLMs (where specific knowledge is used to retrain an LLM and update its parameters), as RAG can leverage a constantly updated knowledge base without requiring regular high-cost retraining (Balaguer et al. 2024). Recently, Radensky et al. (2025) proposed a RAG-based tool for the generation of novel research ideas based on a provided set of research articles. Based on this example, we see that a RAG-enhanced architecture can be used for the use case of idea generation and explore its potential for PIISs.

Objectives and requirements definition

Having identified PII as the most value-adding stage in the BPM lifecycle (Gross et al. 2021) and a lack of artifacts that provide computational support for this stage (Fehrer et al. 2025; Stein Dani et al. 2024), we find the application of GenAI in PII as a promising research area, as GenAI might achieve the level of computational creativity required for the generation of process improvement ideas (Kaplan and Haenlein 2019). Consequently, we defined an initial set of four objectives that our artifact needs to incorporate in its design, which we validated through expert interviews (see Section "Design objectives" for details). The interviewed experts emphasized the need for a human-on-the-loop level of user involvement (Fehrer et al. 2025) and drew our attention to missing and ambiguous elements in the initial set of DOs. This resulted in the development of the updated set of five DOs, on which we elaborate in the following. Table 1 provides an overview of the original and updated sets of DOs and emphasizes changes between these two versions.

DO1. Leverage accumulated BPI knowledge. As identified in the literature review (see Section "Business process improvement" for details), over the last two decades, researchers developed, collected, and structured a substantial body of knowledge on BPI (Fehrer 2023; Koschmider et al. 2019; Reijers and Liman Mansar 2005). Utilizing this existing knowledge in BPI initiatives enables a structured approach to the ideation phase and ensures that existing solutions are not unnecessarily reinvented. Furthermore, some organizations have their own internal collections of BPI best practices and successful case studies, which they want to leverage organization-wide (Fehrer et al. 2024).

Table 1 Initial and updated sets of DOs

Design Objective	Original Version *(before expert interviews)*	Updated Version *(after expert interviews)*	Changes
DO1	Leverage accumulated knowledge about BPI.	Leverage accumulated knowledge about BPI.	No changes.
DO2	Generate process improvement ideas that are relevant to the business process in hand and justify them to facilitate further decision-making.	Generate process improvement ideas that are relevant to the **actual** business process in hand **in its context**.	Emphasized the context aspect. Separated second part (partially transferred to DO5).
DO3	Reduce the expertise and time required to carry out the idea generation process.	Reduce the time and **process improvement expertise** required to carry out the idea generation process.	Specified the type of expertise ('process improvement expertise') that the artifact should reduce.
DO4	Deal with complex BPI initiatives where multiple process inefficiencies are being analyzed in parallel.	Deal with complex BPI initiatives where multiple process inefficiencies are being analyzed in parallel.	No changes.
DO5	*(Introduced after expert interviews)*	Facilitate follow-up steps for human-on-the-loop by explaining and justifying suggested process improvement ideas.	Introduced new DO to explicitly define the level of human involvement in the process.

DO2. Generate context-relevant process improvement ideas. The importance of process context for different BPM stages is increasingly recognized in contemporary literature. Oberdorf et al. (2023) argue that contextual information directly enhances the quality of various BPM stages. Similarly, vom Brocke et al. (2016) emphasize that a one-size-fits-all approach is not suitable for diverse business contexts and propose a framework to define the context of a particular initiative. The PM context taxonomy by Franzoi et al. (2025), distinguishing three contextual levels with three respective dimensions, reinforces the need to consider contextual factors in BPI initiatives. In the area of BPI, Schaschek et al. (2024) observe that a failure to align BPI initiatives with the organizational context reduces the likelihood of their success.

DO3. Reduce time and BPI expertise required. Automation demonstrates the potential to increase human resource utilization and decrease processing time (Denagama Vitharanage et al. 2020). In the field of BPI, the motivation for (semi-)automation originates from the need for increased efficiency, achieved through the minimization of time-consuming, expertise-intensive, and expensive manual tasks (Bergener et al. 2015). Beerepoot et al. (2023) emphasize that automating parts of the BPI process is supposed to accelerate BPI initiatives and make them less dependent on human creativity.

DO4. Handle complex BPI initiatives. Real-world processes often require execution of complex BPI initiatives, i.e. (1) dealing with multiple inefficiencies simultaneously (Li et al. 2023; Tang et al. 2023), (2) dealing with multiple process improvement ideas that impact process performance dimensions in different (sometimes opposite) directions (Reijers and Liman Mansar 2005), (3) dealing with multi-objective BPI scenarios (Tiwari et al. 2010). This complexity underscores the necessity of a coordinated approach, in which all identified process inefficiencies are considered collectively in the context of the specific BPI initiative to formulate a unified process improvement plan.

DO5. Facilitate human-on-the-loop follow-up actions. The debate around fully autonomous systems (human-out-of-the-loop) versus hybrid systems (human-on-the-loop or human-in-the-loop) is becoming increasingly important across various research fields. Recent technological advancements allow for the completion of specific individual tasks fully autonomously (Ribeiro et al. 2021). However, fully autonomous systems are often imperfect, particularly in real-world contexts (Heaven 2019). Consequently, hybrid approaches are gaining popularity, representing a significant trend, with numerous researchers offering guidance on systems involving human-AI collaboration (Amershi et al. 2019; Makarius et al. 2020). A key advantage of hybrid systems is their ability to integrate human knowledge and judgment into AI-based decision-making. This interactive dynamic enhances accountability, reduces bias, and facilitates two-way learning and information exchange (Larrick 2004; Raisch and Krakowski 2021). Furthermore, hybrid systems with human involvement are typically designed with a high level of transparency, substantially enhancing the trustworthiness of these systems (Barredo Arrieta et al. 2020). Following Fehrer et al. (2025, p. 10), we define the level of human involvement as human-on-the-loop, as we design and develop an artifact that "performs the processing autonomously and seeks input from the human process designer at specific points."

Following the taxonomy for PIISs developed by Fehrer et al. (2025), we clearly outline the position of our research within the PIISs landscape. Our *focus of changes* is non-exclusive and might include control flow, resources, and data. We pursue incremental *ambition* with the goal of conducting regular, exploitative BPI initiatives. Following DO2 with a focus on context-relevant process improvement ideas, we use process design data, process execution data, process context, and improvement objectives as *input*. Using the RAG architecture, we plan to combine randomness-based and rule-based *design generation*. *Knowledge acquisition* should be conducted explicitly and collected in a knowledge base. We are not conducting a *design evaluation* explicitly, as the abilities of GenAI for this stage are limited. We foresee a human-on-the-loop *level of human involvement*, in which ideas are generated automatically, and human input is required only at specific points (e.g., evaluation of generated ideas). In the initial version of the artifact, we provide *output* as a non-formalized process design.

Design specification

Based on the identified research gap, formulated DOs, and selected technical approach, we first outline the architecture of our proposed RAG-enhanced LLM-based PIIS in Section "System architecture". To provide a proof of concept (PoC) and demonstrate the technical feasibility of the architecture, we proceed with a minimum viable product (MVP). Following the MVP, we develop an initial software prototype in Section "Software prototype" and refine it based on feedback from the expert interviews (see Section "System architecture and software prototype"). In Section "Initial illustrative example", we illustrate the functionality of the developed PIIS using an academic dataset.

While various terms exist for naming human-computer interaction systems, such as bot, agent, or copilot (Sellen and Horvitz 2024), we select the name Process Improvement Copilot for the proposed PIIS. This naming convention is not merely a linguistic distinction; it reflects a deliberate design choice that shapes user expectations regarding interaction with the system. With the term *Copilot*, we emphasize the selected human-on-the-loop approach, indicating that the system "operates under human direction and oversight" and "… [takes] over tasks as and when needed" (Sellen and Horvitz 2024, p.1).

System architecture

Figure 1 depicts the system architecture of the Process Improvement Copilot. The proposed PIIS generates a list of context-aware process improvement ideas for the identified process inefficiencies based on the relevant knowledge extracted from the knowledge base. Following Wastell et al. (1994), we define process inefficiencies as occurrences in which a business process meets its operational goals but is wasteful in terms of utilized resources.

In Fig. 1, we define the following parts of the architecture. A component is a standalone group of elements and data objects designed to perform a specific function. Elements are basic functional units of the system that perform data processing steps. Similar elements can be grouped into clusters of elements. Data entities are structured data collections (e.g., an event log), individual tables (e.g., a list of process inefficiencies), or variables (e.g., an idea generation prompt) processed and exchanged within the system. User actions are possible user-initiated interactions to control the system or provide input. Relations between components and possible user action flows are marked with solid and dashed arrows, respectively.

The architecture can be broadly 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 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.

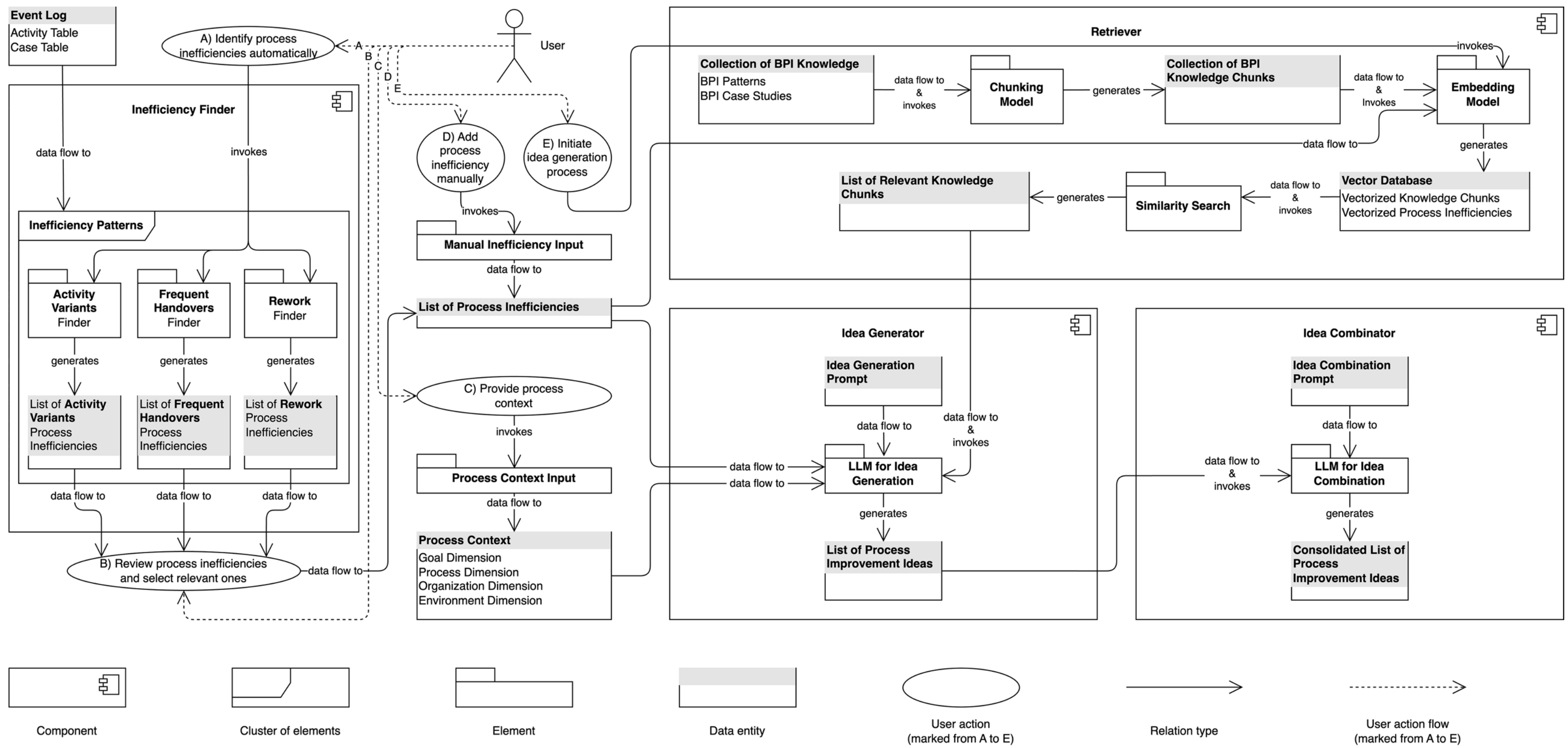

Fig. 1 System architecture of the Process Improvement Copilot

The generation of process improvement ideas within the proposed architecture begins with ingesting the provided *Event Log* into the system (e.g., through uploading an event log file or establishing a connection to the source system). Then, the user should generate a *List of Process Inefficiencies*. They can either follow the way with actions A and B (for automated identification of process inefficiencies) or the way with action D (for manual submission of process inefficiencies). Afterward, the user can follow action C to provide context input. The core idea generation process can be initiated by the user with action E, which invokes the *Embedding Model* and follows the system pipeline further until the *List of Process Improvement Ideas* or the *Consolidated List of Process Improvement Ideas* is generated. Finally, the user can review the suggested process improvement ideas and proceed with further BPI stages, such as idea evaluation and implementation. We design our system in a modular way to enable the flexible replacement, deletion, or addition of individual components and elements. Below, we discuss some of the important design decisions that we made while building system architecture for the Process Improvement Copilot.

Inefficiency Finder: to design a more comprehensive system that covers multiple BPI stages, we include a component that automatically identifies process inefficiencies. This is not the key system component, so we select one of the readily available approaches in research. Among others, we consider (1) BINet, a recurrent neural network architecture that can detect process anomalies (Nolle et al. 2022), (2) a collection of 21 step-by-step analysis templates to systematically identify process inefficiencies (Lashkevich et al. 2023), (3) a deep-learning-based approach to workaround identification (Weinzierl et al. 2022), and (4) SWORD, a pattern-based workaround detection method (van der Waal et al. 2025). We define the following selection criteria. First, the approach should be based on the process execution data to base its output on real-world insights instead of possibly outdated process models. Second, the approach should produce explainable output, i.e., we exclude "black-box" algorithms. Third, the approach should be understandable for the general audience to facilitate human-on-the-loop involvement. These three criteria lead us to the decision to select the approach proposed by Lashkevich et al. (2023) and exemplary implement three of the 21 analysis templates.

Collection of BPI Knowledge: the RAG algorithm is based on a comprehensive knowledge base, which provides relevant pieces of information that are later used for the generation of process improvement ideas. Hence, this collection should include a comprehensive overview of the BPI best practices and case studies. We use the collection of BPI patterns and case studies from the website Process-pattern.app[1] (Fehrer 2023). Proprietary knowledge (e.g., internal BPI case studies or process blueprints) can also be uploaded to the collection.

Chunking Model: chunking is an important step that splits the original documents into smaller pieces. An optimal chunking strategy should find a balance between larger chunks, which capture more context, and smaller chunks, which reduce the noise level in data (Gao et al. 2024). We select semantic-similarity-based chunking, as it splits original documents at the points where the semantic flow significantly changes[2].

Embedding Model: the embedding model transforms semantically chunked documents into high-dimensional vector representations, enabling the system to perform

[1] https://process-pattern.app/

[2] https://python.langchain.com/docs/how_to/semantic-chunker/

vector-based similarity search. The wide range of available embedding models introduces complexity to the selection process. While benchmarks can be employed to evaluate various embedding models, we follow Korade et al. (2024), who demonstrate the efficacy of OpenAI embeddings. Consequently, we design an embedding model using the OpenAI *text-embedding-3-small* model. The *Embedding Model* saves vectorized knowledge chunks in a *Vector Database*.

Similarity Search: similarity search is one of the key elements of RAG, as it determines which pieces of knowledge are later used to generate the answer. We select cosine similarity due to its established use in the field (Han et al. 2012).

LLM for Idea Generation and *LLM for Idea Combination:* given the variety of available LLMs, benchmarking is a justifiable approach for model selection. Kojima et al. (2022) demonstrate the advantages of chain-of-thought prompting in LLMs, hence, we prioritize models with this capability. To maintain consistency, we select and keep the same LLM for the entire project. At the time of selection, GPT-4o was the leading model; therefore, we adopted this model.

The proposed architecture facilitates the achievement of all previously introduced DOs. DO1 (Leverage accumulated BPI knowledge) is achieved by incorporating the *Collection of BPI Knowledge* containing BPI patterns and case studies. DO2 (Generate context-relevant process improvement ideas) is accomplished by utilizing *Event Log* and *Process Contex*t. DO3 (Reduce time and BPI expertise required) is fulfilled, as the system employs a computational algorithm to automate idea generation. DO4 (Handle complex BPI initiatives) is addressed by the *Idea Combinator* component, which consolidates all generated ideas. DO5 (Facilitate human-on-the-loop follow-up actions) is achieved through two primary mechanisms. First, relevant knowledge chunks are extracted from the knowledge base to inform idea generation, ensuring the traceability of idea origins. Second, prompt engineering and LLM configuration are done in a way that minimizes hallucinations and imposes the desired output structure.

Software prototype

Early in the research process, we evaluated the technical feasibility of the proposed system architecture through an MVP in Python. To this end, we formulated multiple 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."*), constructed a basic RAG pipeline, and added seven BPI patterns to the *Collection of BPI Knowledge*. For the simplicity of the MVP, we omitted the *Inefficiency Finder* and *Idea Combinator*.

Internal testing of the MVP indicated that the approach is technically feasible. We observed that the RAG-based prototype could consistently select relevant BPI patterns and generate process improvement ideas that were both higher in quality and more diverse than those generated by the basic LLM (without BPI knowledge base support) that generated its answers based solely on the parametric memory. Furthermore, we noticed that the proposed architecture (i.e., logical steps leading from identifying process inefficiency to selecting relevant pieces of knowledge and, finally, to generating process improvement ideas) resembles human thought processes, potentially enhancing user understanding of the overall concept.

Consequently, we implemented the system architecture (see Fig. 1) as a full prototype. We share the link to the source code of our software prototype as a GitHub repository[3]. The main script of the Process Improvement Copilot (BPI_TOOL_UPDATE.PY) uses the Streamlit library[4] to create an interactive user interface. The users can conduct all possible user actions defined in Fig. 1 in the web application (see Fig. 2). In our later demonstration and evaluation activities, we use event logs stored in a Celonis environment. Hence, one part of the code connects to the Celonis platform using the PyCelonis library[5], allowing the system to access and query process execution data stored in a Celonis environment.

Three Python scripts identify selected process inefficiency patterns in the process execution data (see the *Inefficiency Finder* component in Fig. 1). Each script retrieves information from a Celonis data model and identifies and quantifies inefficiencies. Finally, the script generates a textual description of each inefficiency. We adopt the following process inefficiency patterns from Lashkevich et al. (2023) and demonstrate that they can be identified automatically: *Activity Variants* (identify activities with significant differences between average and median throughput times), *Frequent Handovers* (analyze transitions between activities to identify frequent handovers between different resources), and *Rework* (identify activities executed multiple times in the same case, indicating potential rework and loops).

The system utilizes a knowledge base of BPI patterns and case studies stored as text files in the PROCESS_PATTERNS directory. The files are loaded, split into smaller chunks, and embedded based on the OpenAI embedding model. These actions are coordinated using the LangChain library, which facilitates the development of applications with the integration of LLMs. The embedded chunks are stored in a Chroma[6] Vector Database.

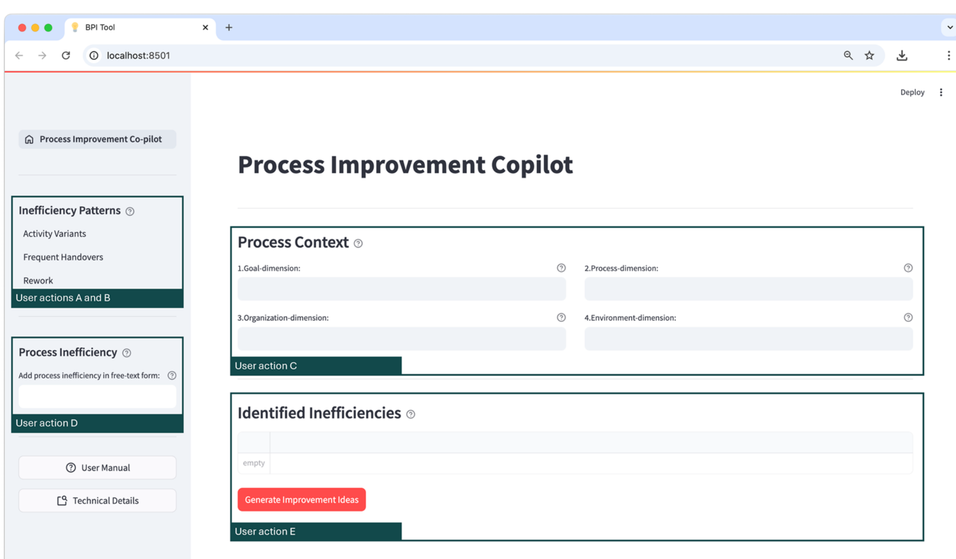

Fig. 2 Software prototype with indications of possible user actions (corresponding to user actions illustrated in the system architecture)

[3] https://github.com/VladimirSmoley/process_improvement_copilot

[4] https://streamlit.io/

[5] https://celonis.github.io/pycelonis/2.12.0/

[6] https://www.trychroma.com/

The GENERATE_IMPROVEMENT_IDEAS() function (corresponds to the user action E in Figs. 1 and 2) takes the identified inefficiencies and process context as input. It retrieves relevant knowledge chunks from the *Vector Database* using the *Similarity Search*. Then, it uses the *LLM for Idea Generation* element to generate specific process improvement ideas (see Fig. 3 for an output example). If multiple process inefficiencies are provided, another part of the GENERATE_IMPROVEMENT_IDEAS() function is invoked. This part runs the *Idea Combinator* component to generate a *Consolidated List of Process Improvement Ideas* based on the *List of Process Improvement Ideas* and *Idea Combination Prompt*.

We formulate prompts for both LLMs using advanced prompting techniques (Schulhoff et al. 2025). The prompt starts by assigning a role to the LLM, instructing it to act as an "experienced assistant specializing in business process improvement." This sets the context and expectations for the LLM's response. Then, the objective and requirements are clearly defined. Additionally, we specify restrictions and the preferred output format. Finally, we add one inefficiency from the *List of Process Inefficiencies*, *Process Context*, and *List of Relevant Knowledge Chunks* provided by the *Retriever* component (for the *LLM for Idea Generation*) and a *List of Process Improvement Ideas* (for the *LLM for Idea Combination*). We instantiate both LLMs with the *temperature* parameter set to 0, indicating that the LLMs should produce the most deterministic and predictable outcome. Generally, the *temperature* parameter controls the trade-off between creativity and predictability of the model's output (Holtzman et al. 2020).

Based on feedback from the expert interviews (see Section "System architecture and software prototype" for details), we refined the prototype as follows. For a full overview of all suggested changes, their priority (*must/should/could/won't/beyond the scope*), and their implementation status, we refer to Table 5 in Appendix 1. We prioritized the suggestions based on the feedback from the interview experts and implemented all the ideas with priority levels *must* and *should*.

During the refinement, we improved the navigation within the app and added tooltips with examples of the expected user input (Changes #1 and #2). We enhanced the LLM prompt to provide a more comprehensive justification for each idea and attached the

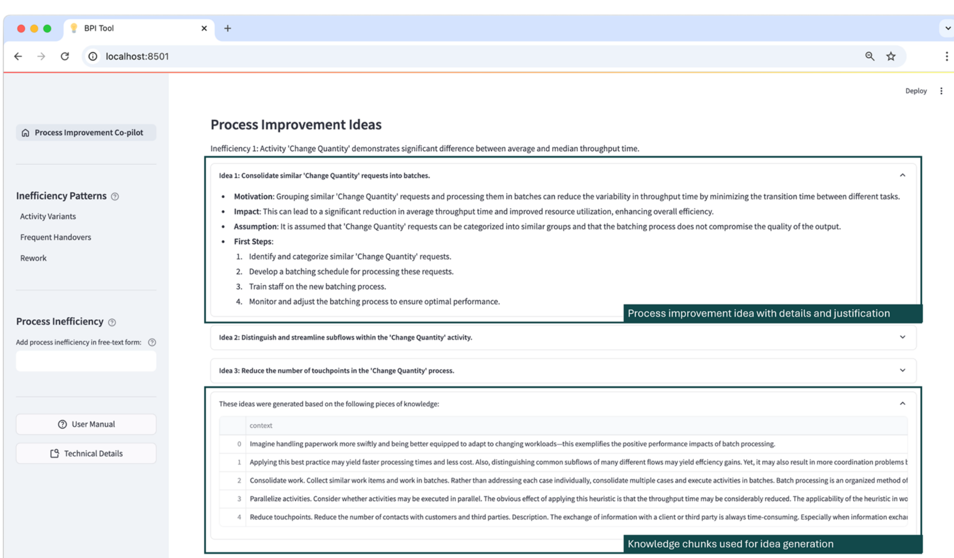

Fig. 3 Output of the Idea Generator component

relevant knowledge chunks used during the idea generation to the output of the Process Improvement Copilot to increase the traceability of the idea generation process (Changes #3 and #4).

Out of Changes #5 and #6, intended to improve user guidance, we implemented Change #5. We preferred a user manual option (Change #5), as it explains to the user all possible actions and suggests a possible approach without strictly guiding them between steps, as they might want to follow a different path, skip individual steps, or repeat some types of analysis.

For Change #7, we changed the output of the *Inefficiency Finder* to provide more data for human-on-the-loop decisions about the relevance of the inefficiency. Changes #8, #9, #10, and #11 represent minor usability improvements, contributing to a more intuitive user interface. Change #12 about providing more information about software architecture covers the interests of more technically advanced users.

We marked Changes #13 and #14 as ongoing, as we uploaded additional pieces of knowledge to the *Collection of BPI Knowledge* and initiated the programming of further inefficiency pattern finders that might be added to the system later. However, the nature of these changes doesn't define a *done* status, as both the knowledge base and the variety of process inefficiency finders can be expanded further.

Other proposed ideas (from Change #15 to Change #26) describe promising changes that can be implemented in the following iterations. We excluded Change #27 about the length of the output, as it was mentioned only by one expert, and implementing it contradicts the feedback from other experts who were satisfied with the length of the text. We also excluded Changes #28, #29, and #30, as they are related to the succeeding stages of the BPI process that we generally exclude from the scope of our research.

Initial illustrative example

To show the technical feasibility of the artifact, we instantiated the software prototype with event log data using a purchase-to-pay event log from the Celonis Academic Alliance Tutorial Package[7] (approximately 5'000 cases, 86 case variants, 20 distinct activities). Table 2 demonstrates the Process Improvement Copilot on the purchase-to-pay event log and serves as an end-to-end example of using the developed artifact in BPI initiatives.

For simplicity reasons, we limit the example to the Rework inefficiency pattern and exemplary proceed with one of the two reported inefficiencies, the repeated execution of the activity 'Change Quantity' in 8.0% of the cases. In 5.0% of the cases, this activity even occurred in direct succession. The relevant knowledge chunks retrieved from the collection of BPI knowledge, represented as individual knowledge chunks in the vector database, include distinguishing common subflows, consolidating work in batches, and using conditional compensations. Additionally, we specify the goal dimension (reducing the process throughput time) and the organizational dimension (multinational OEM) as contextual input for the idea generation prompt compiled by the system.

As the list of process improvement ideas shows, the ideas generated are closely related to the retrieved knowledge chunks and resonate with the specified goal and organizational dimensions. For example, Idea 1 refers to consolidating work in batches, justified by a reduction in process throughput time and an optimized resource utilization. Similarly, Idea 2 originates from the knowledge chunk suggesting the conditional compensations, whereas Idea 3 incorporates the best practice of reducing time-consuming touchpoints.

[7] https://www.celonis.com/academic-alliance/

Table 2 End-to-end example of the Process Improvement Copilot generating context-specific process improvement ideas for an automatically identified inefficiency

Process Improvement Copilot Elements	**Example**
Event Log	Purchase-to-pay event log from the Celonis Academic Alliance Tutorial Package (approximately 5'000 cases, 86 case variants, 20 distinct activities)
List of Rework Process Inefficiencies	- Activity 'Change Quantity' is executed multiple times in 8.0% of cases (in 5.0% of cases it's executed in loop). - Activity 'Cancel Goods Receipt' is executed multiple times in 7.0% of cases (in 7.0% of cases it's executed in loop).
List of Process Inefficiencies	Activity 'Change Quantity' is executed multiple times in 8.0% of cases (in 5.0% of cases it's executed in loop).
Collection of BPI Knowledge	Collection of BPI patterns and case studies (Fehrer 2023)
Collection of BPI Knowledge Chunks	*(omitted for conciseness)*
Vector Database	*(omitted for conciseness)*
List of Relevant Knowledge Chunks	- Consolidate work. Collect similar work items and work in batches. Rather than addressing each case individually, consolidate multiple cases and execute activities in batches … *(shortened for conciseness)* - Variations. This pattern can be used in a custom way by using conditional compensations. For example, the compensation process or activity is only performed at every tenth initiation of the original process … *(shortened for conciseness)* - Reduce touchpoints. Reduce the number of contacts with customers and third parties. Description. The exchange of information with a client or third party is always time-consuming … *(shortened for conciseness)* - … *(shortened for conciseness)*
Process Context	**Goal Dimension:** Reduce process throughput time. **Organizational Dimension:** Multinational OEM.
Idea Generation Prompt (part of the algorithm; no additional user input required)	**Role:** You are an experienced assistant specializing in business process improvement. **Objective:** Your task is to identify and suggest three specific process improvement ideas for the given process inefficiency in the given process context. **Requirements:** Clarity …, Relevance …, Specificity …, Justification …, Impact …, Implementation …, Step-by-step solution …, Knowledge-driven … *(shortened for conciseness)* **Restrictions:** … *(shortened for conciseness)* **Inputs:** Identified Process Inefficiency: {process_inefficiency}; Knowledge: {knowledge}; Process context: {process_details} **Output:** … *(shortened for conciseness)*
List of Process Improvement Ideas	**Idea 1:** Implement batch processing for the 'Change Quantity' activity. *Motivation*: Consolidating similar 'Change Quantity' requests and processing them in batches can reduce the frequency of the activity being executed multiple times. *Impact*: This can significantly reduce process throughput time by minimizing transitions and optimizing resource utilization. *Assumption*: The 'Change Quantity' requests can be grouped without negatively impacting other process requirements or customer satisfaction. *First Steps*: Identify and categorize similar 'Change Quantity' requests, determine optimal batch sizes, and develop a batching schedule. **Idea 2:** Introduce conditional compensations for the 'Change Quantity' activity. … *(shortened for conciseness)* **Idea 3:** Reduce touchpoints in the 'Change Quantity' activity. … *(shortened for conciseness)*

Demonstration and evaluation

To formalize and quantify the utility of the Process Improvement Copilot in artificial and naturalistic environments, we initiate a series of evaluation activities. Considering the overall DSR-based structure of this paper, we developed our evaluation strategy in line with the Framework for Evaluation in DSR (Venable et al. 2016), which recommends transitioning from a formative to a summative purpose. Furthermore, we apply evaluation methods proposed by Sonnenberg and vom Brocke (2012), such as expert interviews, surveys, benchmarking with related approaches, and experiments with a prototype. We evaluated the importance and completeness of our DOs in 13 semi-structured expert interviews (Section "Design objectives"), which led to the refinement of the DOs described in section "Objectives and requirements definition". Using the refined DOs, we assessed the novelty of the Process Improvement Copilot through a feature comparison with related approaches (Section "Competing artifact analysis"). With the same panel of experts, we evaluated the system architecture and the initial version of the prototype (Section "System architecture and software prototype"). Finally, we investigated the real-world applicability of the Process Improvement Copilot in a workshop at a multinational technology conglomerate (Section "Application in a real-world setting").

Consequently, we gathered a diverse panel of 16 experts from academia and industry (see Table 6 in Appendix 2). Our key criterion for experts was a total of at least three years of research and practical experience in BPM with a particular focus on BPI. Academic experts are doctoral candidates and researchers with completed doctoral degrees from various German research institutions. Most of them additionally bring industry expertise collected through consulting projects. The industry experts are employees in different industries and, to a large extent, responsible for BPI initiatives in their organization. A major advantage of expert interviews is the flexibility regarding possible question adjustments based on the interviewee's experience and competence (Magaldi and Berler 2020).

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 DOs, (4) an introduction of the system architecture, (5) a demonstration and evaluation of the software prototype instantiated on the 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.

Overall, 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. Expert 11 saw the potential for the Process Improvement Copilot to serve as a "sparring partner," providing an initial version of the process improvement ideas, thus reducing the need to start from scratch. 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.

Design objectives

Figure 4 shows the experts' agreement with the importance of each DO for the initial and subsequently updated set of DOs. The discussion of the initial set of DOs during the interviews led to the following insights:

DO1 (Leverage accumulated BPI knowledge). Several interviewees emphasized the potential benefits of utilizing a common knowledge base of BPI practices at the organizational level as well as at the research community level. For example, Expert 11 supported this view, suggesting that leveraging existing knowledge is a significant advantage of the RAG-based PIIS compared to human process designers who might recall only a limited amount of information when conducting BPI initiatives. However, several experts raised concerns about the potential limitations of relying solely on past knowledge. For example, Expert 9 questioned whether deep knowledge of all BPI methods is always necessary, suggesting that a strong understanding of the specific process might be sufficient.

DO2 (Generate context-relevant process improvement ideas and justify them). 14 out of 16 experts ranked this DO as *extremely important*. Expert 12.1 reiterated this point, stating that high-level process-unrelated ideas cannot be implemented. The interviewees also highlighted the need to justify the generated ideas to facilitate their further exploration and implementation. Some experts encouraged us to reconsider whether two parts of the DO are non-separable or should be considered independently (see Section "Objectives and requirements definition" for details).

DO3 (Reduce expertise and time). Half of the experts considered this DO *extremely important* and emphasized that our artifact would not create extra value without fulfilling this criterion. However, while agreeing that time reduction is an important objective, many experts stated that reducing the general expertise (not specifically BPI expertise) might be neither desirable (considering the possible loss of human control in the process) nor feasible (considering the state-of-the-art of the technology). We reconsidered this aspect and adapted this DO accordingly (see Section "Objectives and requirements definition" for details).

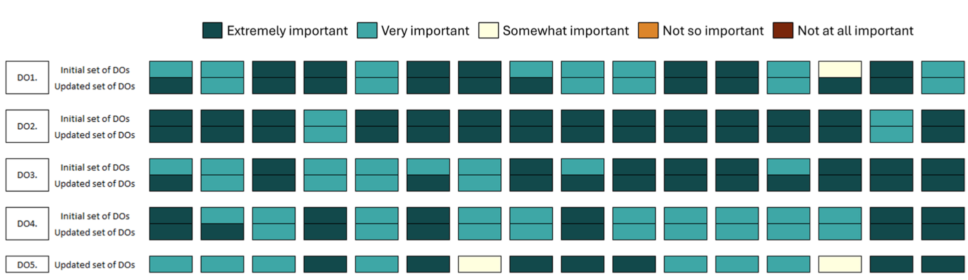

Fig. 4 Evaluated importance of the initial set of DOs (based on the interview results) and the updated set of DOs (based on the survey results)

DO4 (Handle complex BPI initiatives). Six experts ranked this DO as extremely important, emphasizing that a desirable new process model can only be achieved by jointly considering all inefficiencies within one process. At the same time, Expert 1 and Expert 4 expressed an opinion that considering multiple inefficiencies might go beyond the scope of a basic solution and suggested focusing on the generation of high-quality ideas for individual inefficiencies first.

Most experts agreed that for the suggested definition of the problem scope, the set of DOs was complete. However, two interviewees (Expert 11 and Expert 13) pointed out the lack of a DO describing the role of humans in interaction with the artifact. They emphasized the particular importance of such a DO, especially for artifacts that are not fully autonomous.

Considering this comment and the ambiguity of the definition of *expertise* (see DO3 above), we revised the initial set of DOs (see Section "Objectives and requirements definition" for details). To guarantee the rigor of the research project, we decided to specifically evaluate the updated set of DOs. With this additional step, we intended to show that it was sensible to implement the changes to the set of DOs and that the updated set of DOs satisfies the requirements of importance and completeness. Therefore, each interview participant was invited to participate in a survey assessing the importance of the updated DOs. The invitation included a concise summary of the changes and a justification for each modification. We present a comparison of the initial and updated rankings for each DO in Fig. 4.

The ranking of DO2 remained constant (i.e., none of the experts changed their ranking), indicating that the simplification from the initial to the updated version of the DO retained the key idea. It was still considered *extremely important* by the majority of the experts. Based on the increased evaluated *importance* (four experts revised their ranking of DO3 from *very important* to *extremely important*), we can conclude that clarifying the type of required expertise enhanced the clarity of DO3. The new DO5 was also generally considered important by the experts (seven ratings of *extremely important*, seven of *very important*, and two of *somewhat important*).

Notably, the experts did not have access to their responses from the prior evaluation activity. Combined with the time interval between these two evaluation activities, this reason might explain minor observed shifts in the ranking of DO1 and DO4 despite these DOs remaining unchanged.

Competing artifact analysis

To evaluate the novelty of the Process Improvement Copilot, we conducted a competing artifact analysis. The competing artifact analysis aims to position our artifact among similar (semi-)automated approaches and outline its distinguishing characteristics. To select competing artifacts, we refer to the overview of the BPI methods and tools in the literature review (see Section "Business process improvement" for details) and define two exclusion criteria. First, the artifact should be a PIIS (i.e., provide computational support), which excludes classical manual methods for generating process improvement ideas (e.g., Reijers and Liman Mansar 2005; Zellner 2013). This exclusion is motivated by the focus of our research on the (semi-)automation of the BPI process. Second, the artifact should be instantiated as a prototype, which excludes conceptual papers on the automation of the BPI stage, such as Schuh et al. (2021), and ongoing research projects without completed demonstration and evaluation, such as Harl et al. (2024). This criterion is motivated by the possibility that ongoing research initiatives might evolve the architecture of their respective solutions or remain entirely conceptual. Based on these two exclusion criteria, we compiled a list of eight competing artifacts.

Similar to Egger et al. (2025), Kecht et al. (2023), and van Dun et al. (2023), we used the fulfillment of the updated set of DOs as comparison criteria in the competing artifact analysis. Table 3 summarizes the results of the competing artifact analysis. The Harvey balls indicate whether a DO is considered fully (●), partially (◐), or not all (○) by the respective artifact, followed by a justification of the assessment. To mitigate subjective bias, the initial assessments and justifications from one of the authors were verified by the other authors.

The competing artifact analysis reveals that for most of the artifacts, the scalability to new BPI knowledge is limited due to the manual pre-processing steps involved. Some artifacts refrain entirely from using BPI knowledge in the pursuit of creativity (Afflerbach et al. 2017; van Dun et al. 2023). Regarding DO2, most of the compared artifacts insufficiently cover context factors, limiting their ability to generate contextually relevant improvement ideas. All artifacts at least partially reduce the time and BPI expertise required to generate process improvement ideas (DO3). Explicit handling of multiple process inefficiencies (DO4) is generally neglected in existing artifacts, indicating a common limitation in addressing complex BPI initiatives comprehensively. Regarding DO5, the compared artifacts typically lack explicit support for human-centered follow-up actions, showing limited guidance or actionable output for practitioners to implement improvements.

Moreover, the Process Improvement Copilot satisfies the design requirements and DPs for PIISs introduced by Moder et al. (2025). It increases the effectiveness of PII by generating process designs that solve identified process inefficiencies (design requirement 1), increases the efficiency of PII by automating the generation of process improvement ideas (design requirement 2), while maintaining human control in the selection of relevant process improvement ideas and supporting process designers with explanations for the generated process improvement ideas (design requirement 3).

Through the inclusion of the BPI knowledge, process context, and event log, the Process Improvement Copilot satisfies both Input-DPs *"Exploit existing PII knowledge"* and *"Leverage multiple sources and a variety of diverse data"*. Out of the Throughput-DPs, we focus on the DP3 (*"Automatically generate and suggest process design proposals"*) through using LLM for automated generation of the process improvement ideas, DP4 (*"Target several process perspectives"*) through generation of process improvement ideas for different process perspectives such as control flow or resources, DP5 (*"Account for process context factors"*) through explicit inclusion of process factors, DP7 (*"Conduct a multi-objective process assessment"*) through option to include multiple process improvement objectives, and DP8 (*"Enable interactive and iterative PII"*) through step-by-step guidance within the Process Improvement Copilot. Among Output-DPs, the Process Improvement Copilot satisfies DP10 (*"Provide adaptable, atomic process design proposals"*), DP11 *("Provide information for process decision-making on different levels of granularity"*), and DP12 *("Provide explanations for process design proposals"*) through supporting every generated process improvement idea with reasoning and suggested next steps. Moreover, Process Improvement Copilot covers DP14 (*"Ensure interoperability with the existing system landscape and connected BPM activities"*) through the possibility of accessing event logs directly from Celonis.

Table 3 Competing artifact analysis

Approach	DO1. Leverage accumulated BPI knowledge.	DO2. Generate context-relevant process improvement ideas.	DO3. Reduce time and BPI expertise required.	DO4. Handle complex BPI initiatives.	DO5. Facilitate human-on-the-loop follow-up actions.
Process Improvement Copilot *(current research)*	● Use of BPI patterns, best practices, and case studies. Scalability to other knowledge is possible	● Event log is used to understand relevance of an idea to the context. Additional process context can be provided manually	● Time and BPI expertise for the specific step of idea generation are reduced	◐ Explicit handling of multiple process inefficiencies together	● Comprehensive overview of the assumptions, estimated impact, and next steps are provided
Afflerbach et al. (2017)	○ No BPI knowledge explicitly used	◐ Context is limited to input matrices. No use of event data or process documents	◐ Time for the whole BPI process is reduced. Providing input and understanding output are expertise-intensive steps	○ Individual process inefficiencies are not recognized	◐ Estimation of impact of every redesign option is provided. No justifications are provided
Bergener et al. (2015)	◐ Use of formalized process weaknesses. Scalability to other knowledge is limited	◐ Context is limited to semantic process model. No use of event data or process documents	● Time and BPI expertise for the specific step of problem recognition are reduced	○ No explicit handling of multiple process inefficiencies together	○ No explicit info is provided. Focus is solely on weakness identification
Fehrer et al. (2022)	◐ Use of BPI patterns. Scalability to other knowledge is limited	◐ Context is limited to process model. No use of event data or process documents	◐ At high automation levels (available only for two BPI patterns), time and BPI expertise for the whole BPI process are reduced	○ No explicit handling of multiple process inefficiencies together	◐ At the highest automation level (available only for one BPI pattern), simulation and interpretation are provided
Fehrer et al. (2024)	◐ Use of (formalized) BPI patterns. Scalability to other knowledge is limited	◐ Event log is used to automatically check relevance of an idea to the context in a rule-based way. No use of process documents	◐ Time and BPI expertise for the specific step of idea generation are reduced. Translation of BPI patterns into rulesets is an expertise-intensive step	○ No explicit handling of multiple process inefficiencies together	◐ Suitability score (with detailed info on components) is provided
Mustansir et al. (2022)	○ No BPI knowledge explicitly used	◐ Context is limited to end-user feedback on the process. No use of event data or process documents	◐ Time for input analysis is reduced	○ No explicit handling of multiple process inefficiencies together	○ No explicit info is provided. Focus is solely on extraction of redesign suggestions

Table 3 (continued)

Approach	DO1. Leverage accumulated BPI knowledge.	DO2. Generate context-relevant process improvement ideas.	DO3. Reduce time and BPI expertise required.	DO4. Handle complex BPI initiatives.	DO5. Facilitate human-on-the-loop follow-up actions.
Niedermann and Schwarz (2011)	◐ Use of formalized optimization patterns. Scalability to other knowledge is limited	● Operational and event data is used to automatically detect inefficiencies and propose improvement ideas	● Time and BPI expertise for the specific step of idea generation are reduced	○ No explicit handling of multiple process inefficiencies together	◐ Expected effects of idea implementation are provided
van Dun et al. (2023)	○ No BPI knowledge explicitly used	◐ Event log (with the focus on positive deviance) is used to generate improvement ideas. Additional process context cannot be provided	● Time and BPI expertise for the specific step of idea generation are reduced	○ Individual process inefficiencies are not recognized	○ No explicit info is provided. Lack of explainability and understandability due to use of neural networks
Zemni et al. (2016)	○ No BPI knowledge explicitly used	○ No use of process models, event data or process context	● Time and BPI expertise for the specific step of process model generation are reduced	○ Individual process inefficiencies are not recognized	○ No explicit info is provided. Focus is solely on the composition of process fragments

System architecture and 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 2012). 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 Fig. 5). 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.

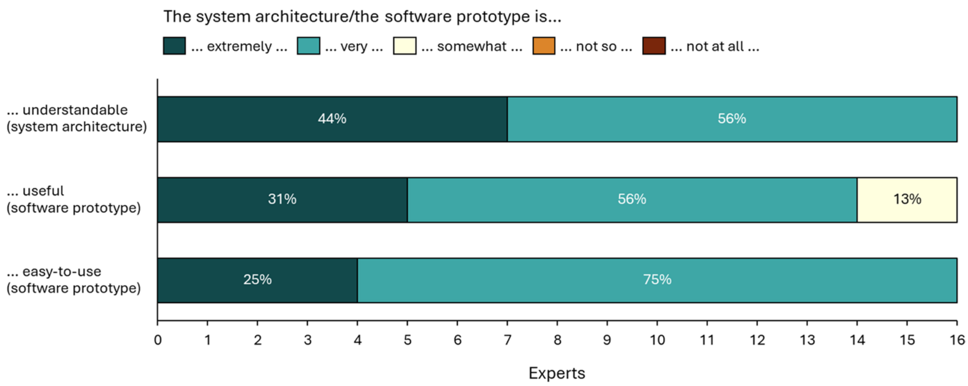

Fig. 5 Evaluated *understandability*, *usefulness*, and *ease-of-use* of the system architecture and the software prototype (based on the interview results)

Regarding the *usefulness* of the software prototype, we observed two groups of experts. The former group of experts 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 of experts emphasized the inseparability of the *usefulness* of the software prototype from the *usefulness* of the generated process improvement ideas in their opinion (e.g., Expert 4 commented, “It looks very promising, but I’m not aware of the actual [business] process, so I cannot really assess.”). We fully understand this concern and cover it by conducting another naturalistic evaluation activity on an event log that is familiar to the evaluation participants (see Section “Application in a real-world setting”).

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). We comprehensively addressed these comments in a subsequent development iteration (see Section “Software prototype” and Table 5 in Appendix 1 for details).

In line with DO3, 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 Fig. 6). 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. Expert 13 shared from their experience that collecting background knowledge and getting familiar with relevant BPI practices might take up to several weeks in real-world cases, which means a significant time reduction coming from such a PIIS that can generate process improvement ideas based on the BPI knowledge within minutes.

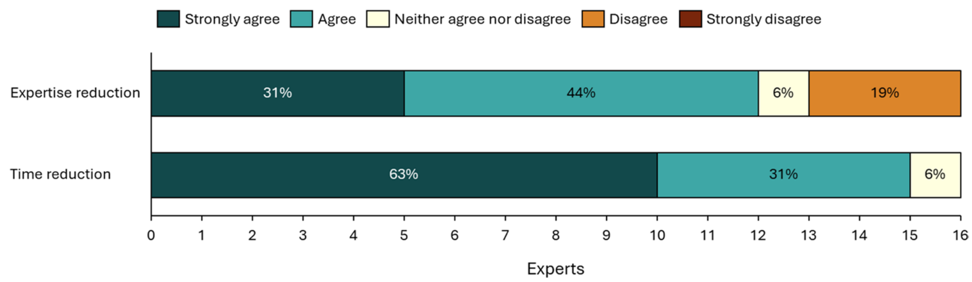

Fig. 6 Evaluated *expertise reduction* and *time reduction* for the idea generation process attributable to artifact use (based on the interview results)

The opinions on the reduction of required expertise were more varied. While confirming that the software prototype significantly reduces the need for specialized knowledge (e.g., Expert 9 stated, "I don't need to know anything about process improvement patterns and so on because everything was served to me."), many experts simultaneously emphasized that process-specific expertise remains crucial (e.g., Expert 9 observed, "I still need the process expertise because I will only get an assumption and the justification and so on. But then I still have to know the process really well").

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 and improvement ideas, which we discussed in detail in Section "Software prototype". Finally, Expert 4 raised an important concern regarding the costs of developing and operating this GenAI-based system on large datasets. We agree that it is crucial to consider this aspect when evaluating the benefits and costs of the proposed PIIS (and any other GenAI-based PIIS) in real-world BPI initiatives.

Application in a real-world setting

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 in industry, infrastructure, transport, and healthcare. The 90-minute workshop was logically separated into two parts. In the first part, after a short 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. First, we instantiated the software prototype on the purchase-to-pay event log from the conglomerate's supply chain management (SCM) department. Then, the participants were asked to identify some process inefficiencies and generate a set of process improvement ideas with the help of the Process Improvement Copilot. After completing this task, the participants evaluated ideas suggested by the software prototype. We encouraged the participants to ask clarifying questions and share their thoughts and feedback throughout the workshop.

We used the event log from the purchase-to-pay process for indirect materials (approximately 2.5 million cases; approximately 210'000 case variants; 70 distinct activities). In a simplified overview, an optimal process begins with creating a purchase order. Following necessary review procedures, the purchase order is transmitted to the supplier. A goods receipt is recorded upon arrival. Once the invoice is received, it is paid, and the case is closed. As expected in a real system with real data, the case demonstrates diverse variations, including additional activities (such as changing the delivery date or price, sending reminders, and setting or releasing blocks) and non-optimal flows (e.g., loops). All participants were familiar with the process. The workshop was recorded and transcribed to extract the thoughts and feedback that participants shared directly in the workshop.

For the workshop, we invited eight participants across various departments and functions. When selecting workshop participants, we focused on their *domain expertise* and *seniority*. We invited participants either with BPM/BPI *expertise* (due to the domain of our PIIS) or SCM *expertise* (due to the domain of the purchase-to-pay process used in the practical part of the workshop). Some experts are working on the intersection of these domains (e.g., Senior Manager for Process Optimization and Digitalization with 30+ years of relevant experience or Lead Architect for a major process transformation and harmonization project with 10+ years of relevant experience). Considering that the artifact should support users from different levels of *seniority*, we invited three participants with one to four years of relevant experience, three participants with seven to ten years of relevant experience, and two senior-level participants with more than 15 years of relevant experience.

Generally, the workshop demonstrated that the Process Improvement Copilot provides value in real situations. Despite being critical at some points regarding input and output quality, the experts recognized the key potential of the proposed PIIS which is providing computational support for the generation of process improvement ideas.

At the end of the first part of the workshop, the experts evaluated the system architecture's *understandability* and the software prototype's *usefulness* and *ease-of-use*. To ensure comparability of the results, we kept the definitions of the criteria the same as in Section "System architecture and software prototype". Although we updated the software prototype in line with the changes proposed by the interview experts, a lower ranking across all three criteria was observed in the workshop setting. Despite the decrease, the rankings remained high, with the majority of experts evaluating the system architecture as *very understandable*, and the software prototype as *very useful* and *very easy-to-use* (see Fig. 7). We assume that the lower ratings might be attributed to two primary factors: 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 the differing expectations of

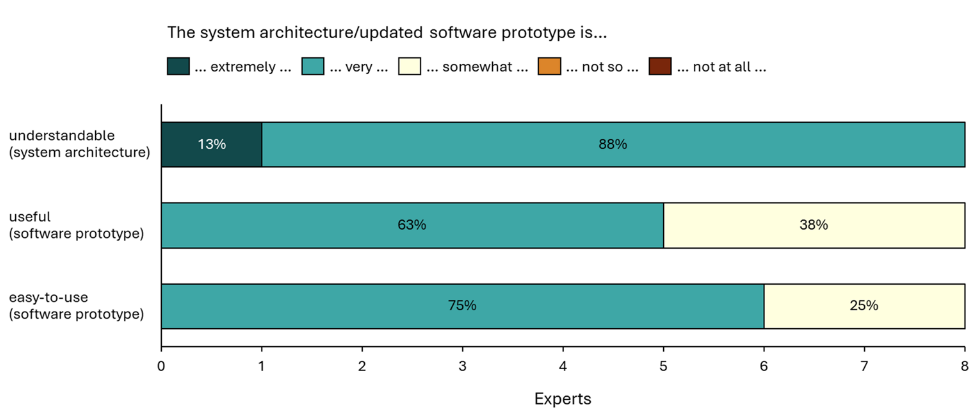

Fig. 7 Evaluated *understandability*, *usefulness*, and *ease-of-use* of the system architecture and the updated software prototype (based on the workshop results)

Table 4 Process improvement ideas generated by the updated software prototype (results of the workshop)

Identified Process Inefficiency	Process Improvement Idea
Activity "Send Purchase Order" is executed multiple times in XX% of cases (in XX% of cases it's executed in loop). *(automatically identified)*	• Implement a sequential checking strategy for the "Send Purchase Order" activity to prioritize conditions with high termination probability and low effort. • Distinguish and separate different case types for the "Send Purchase Order" activity to streamline processing.
Consequent activities "Pay Invoice" and "Set Final Delivery Indicator (ERP)" are executed by different resources in XX% of cases. *(automatically identified)*	• Combine "Pay Invoice" and "Set Final Delivery Indicator (ERP)" into a single role. • Implement a workflow management system to streamline "Pay Invoice" and "Set Final Delivery Indicator (ERP)".
Activity "Change Price" is only executed once per week. Until then all cases are blocked. *(manually provided)*	• Establish a flexible "Change Price" activity schedule based on case urgency. • Introduce a daily "Change Price" activity to reduce the frequency of batch processing".

workshop participants (while the interviewees in the previous sections 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).

As previously outlined, the second part of the workshop replicated a process improvement workshop. Each participant ranked the generated ideas according to criteria derived from the DOs. Two perspectives were employed: an individual perspective, ranking each idea separately, and a collective perspective, ranking the output of the Process Improvement Copilot holistically. We applied the individual perspective for questions where variations in the ratings across different ideas were possible. These were the questions corresponding to DO2 (which requires that ideas should be relevant to the specific process and its context) and DO5 (which requires that ideas facilitate follow-up human-on-the-loop actions with justifications), as the quality of ideas and justifications may vary across different ideas. The collective perspective was applied to the questions where, by design, uniform ratings across all ideas were expected. These were the questions corresponding to DO1 (which requires that ideas should leverage BPI knowledge), DO3 (which requires that the artifact should reduce the required time and BPI expertise), and DO4 (which requires that the artifact should adequately consider cases with multiple process inefficiencies).

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. Some examples of the process inefficiencies and process improvement ideas (confidential information is concealed) are provided in Table 4.

Each of the eight workshop participants ranked six process improvement ideas, resulting in 48 rankings (see Fig. 8). 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). The experts frequently attributed the modest ratings to the necessity of further investigation. Without a deeper examination of both the identified process inefficiency and the corresponding process improvement ideas, determining their relevance remains challenging. Nonetheless, the experts acknowledged the value of a system that generates such ideas and encourages subsequent investigation. These levels of idea relevance and actionability

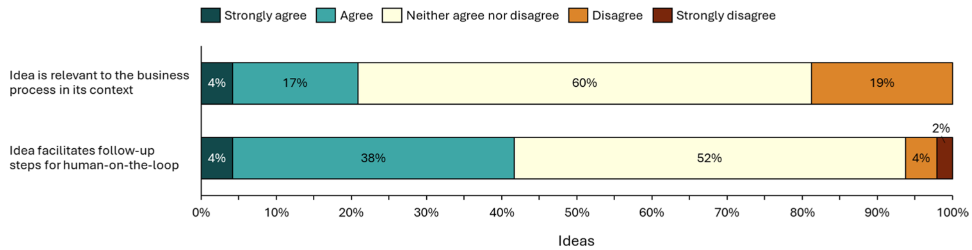

Fig. 8 Evaluated *relevance* and *actionability* of the generated process improvement ideas (based on the workshop results)

satisfy the expectations of the real users: one workshop participant stated verbatim, "[These ideas] are something that we can look at. If [the tool] comes up with 50 ideas, we might find five which are valid to follow up." This perspective was shared by other workshop participants and aligns with the opinions expressed by some interviewees in Section "System architecture and software prototype", reinforcing the role of our PIIS not as a fully automated solution but rather as computational support for human-on-the-loop process improvement. In line with this observation, 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. 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.

Figure 9 shows the experts' agreement with the statement that the Process Improvement Copilot leverages BPI knowledge to generate process improvement ideas (five experts *agree* or *strongly agree* with this). Furthermore, we showed that using the Process Improvement Copilot reduces the time and BPI expertise required to generate process improvement ideas. Consistent with the interview findings, the experts noted that time reduction is more readily achievable than BPI expertise reduction, as users must still possess a thorough understanding of the preceding and succeeding stages of the BPI process. The evaluation also revealed the potential for improvement analyzing multiple inefficiencies (with only half of the experts agreeing that the analysis is adequate).

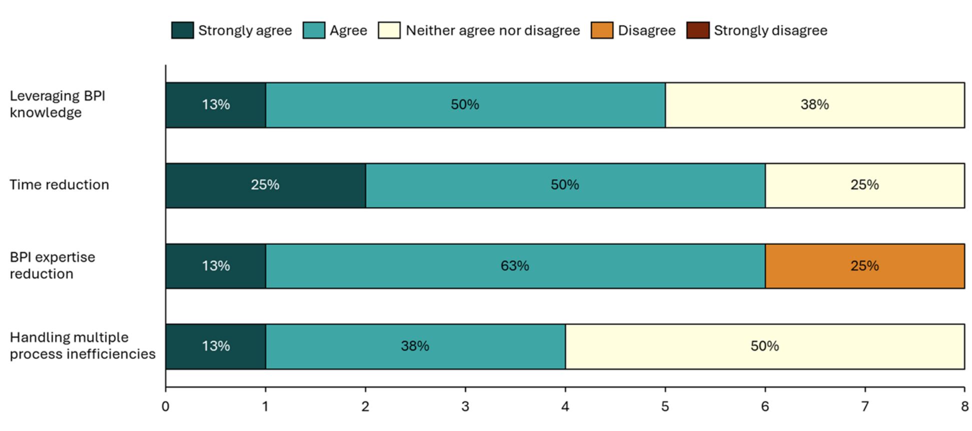

Fig. 9 Evaluated *leveraging of BPI knowledge*, *time reduction*, *BPI expertise reduction*, and *handling of multiple process inefficiencies* for the idea generation process attributable to using the Process Improvement Copilot (based on the workshop results)

Apart from collecting quantitative rankings, we also focused on the discussions and qualitative feedback. One recurring topic was the crucial role of data quality. Several experts highlighted limitations related to the quality of input data (event log), particularly concerning the accuracy of automated inefficiency recognition. This underscores the need for a human-on-the-loop approach to validate the identified inefficiencies before generating process improvement ideas. Several experts expressed the concern that the Process Improvement Copilot is currently focusing on addressing the symptoms of process inefficiencies rather than their underlying root causes. One expert stated, "I would also have to go back and do an interview step to understand why and whether [we have this inefficiency]." The potential to enhance the RAG knowledge base by uploading company-specific documents (e.g., company standards, process blueprints, case studies) was also discussed. This could allow the Process Improvement Copilot to draw upon a richer, more contextually relevant set of information when generating improvement ideas. Further development suggestions focused on refining the output by incorporating mechanisms for filtering, simulating, and prioritizing improvement ideas. The need for a rule-based system to exclude irrelevant or non-compliant suggestions was highlighted. Conversely, another expert viewed such suggestions as a non-critical issue and appreciated the ability (due to the inherent creativity) to challenge existing rules and encourage users to reflect on the validity of the current rules. The experts acknowledged the value of the generated ideas as stimuli, even if many of them were not relevant. This feedback indicates that our proposed PIIS could serve as a valuable brainstorming tool, prompting consideration of a broader range of solutions. Still, the need for iterative refinement of ideas and some built-in feedback mechanisms were emphasized by another expert. The possibility of incorporating reinforcement learning through a voting system was suggested to achieve this iterative improvement.

Conclusion

Summary and contribution

BPI is arguably the most value-adding stage of the BPM lifecycle (Gross et al. 2021). However, its demands on time, BPI expertise, creativity, and its complexity result in limited computational support (Beerepoot et al. 2023; Röglinger et al. 2021). Recent advancements in AI have created opportunities to automate tasks traditionally requiring human intelligence (Rai et al. 2019). For computational support of the BPI stage, the creative capabilities demonstrated by GenAI are of particular interest. This led us to the research question how a RAG-enhanced LLM-based PIIS that supports the generation of process improvement ideas in BPI initiatives can be designed.

We structured our research in the spirit of the DSR paradigm (Peffers et al. 2007). After identifying the research gap, we formulated and evaluated five DOs, which we later incorporated into the development of the system architecture of the Process Improvement Copilot – a RAG-enhanced LLM-based PIIS that supports idea generation in BPI initiatives. The RAG-enhanced architecture compensates for the traditional weaknesses of LLM-based systems, such as hallucinations and lack of control over the origin of the output, by introducing an additional system component that retrieves relevant chunks of knowledge from the knowledge base, thus providing the LLM with a non-parametric memory for output generation (Lewis et al. 2020). Based on that, we implemented a software prototype in Python and conducted a series of demonstration and evaluation activities.

To illustrate the developed functionalities, we applied the software prototype to an academic dataset of a purchase-to-pay process. We assessed the importance and completeness of the DOs through 13 interviews with 16 academic and industry experts. Driven by some updates in the DOs after the interviews, we conducted a survey to validate the importance of the updated DOs. A competing artifact analysis benchmarked the Process Improvement Copilot against eight competing artifacts and demonstrated its competitive advantages. With the same panel of experts, we also evaluated the research gap, the system architecture and an initial version of the software prototype. The experts recognized the importance of the problem area and agreed that our proposed PIIS reduces the time and BPI expertise required to generate process improvement ideas. Finally, we conducted a process improvement workshop with BPM experts and domain experts from a multinational technology conglomerate to evaluate the prototype's value in a real-world setting. The majority of the workshop participants found the Process Improvement Copilot *very understandable*, *very useful*, and *very easy-to-use*. Furthermore, the interview experts and workshop participants frequently mentioned that receiving even one valuable idea out of five or ten can be highly beneficial. Therefore, we interpret these results as promising.

This research contributes to theory and practice. 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, thus advancing the discovery of PIIS architectures based on computational creativity. This contribution stimulates further scientific discourse at the intersection of computational creativity and BPI, thus contributing to the interdisciplinary field of process science. Second, we demonstrate the practical value of the Process Improvement Copilot in the workshop at a multinational technology conglomerate, allowing us to draw conclusions on its usefulness in real-world scenarios. Third, we encourage further exploration of the opportunities to leverage GenAI in PIISs by openly sharing our code via the GitHub repository. We organize the code in a component structure that enables the flexible replacement or update of individual code parts.

Implications for research and practice

While the implementation of the software prototype primarily relies on established concepts (e.g., inefficiency patterns proposed by Lashkevich et al. (2023) or a chunking model for the retriever offered as a function in the LangChain Python library), the orchestration of various components and elements proved non-trivial. The primary focus of this research project was achieving a functional implementation of the PIIS rather than optimizing for a perfectly fine-tuned RAG configuration for the use case of idea generation in BPI initiatives. Therefore, we see potential to enhance the quality of the system output by implementing more advanced elements and components that we identified during our research.

The results of the demonstration and evaluation activities indicate that our PIIS can effectively support users in generating process improvement ideas. The interview experts and workshop participants perceived the artifact represented by the prototype as intuitive, as its reasoning process combines a deep understanding of process inefficiencies with existing BPI knowledge while also incorporating elements of creative exploration and research, thus resembling a human way of generating process improvement ideas. Frequently, the experts noted that, in practice, only a small fraction (usually between 10% and 20%) of the initially generated process improvement ideas prove feasible and viable, which is typically determined only during the succeeding idea evaluation stage of BPI. This observation underscores the value of more detailed output, as the justifications and relevant knowledge chunks used for idea generation offer valuable insights and starting points for further analysis.

While the research demonstrates the potential of retrieval-augmented idea generation in BPI, a scalable application of the proposed PIIS in organizations requires further development of the prototype. This includes extending the functionality of the already involved LLM to a conversational format (instead of single-message output) to support deeper analysis of the specific idea in human-AI interaction. Furthermore, integrating preceding and succeeding BPI stages into the PIIS would ensure a more holistic approach. This might involve incorporating other technologies into the system, as GenAI is not a universally optimal technology for all BPI stages. Finally, incorporating more BPI knowledge (e.g., case studies with quantified results) and more context knowledge (e.g., business rules) can potentially improve the quality and relevance of the generated process improvement ideas.

Finally, to provide more guidance on potential use cases for the proposed PIIS, we discuss three application scenarios of the Process Improvement Copilot. First, a process analyst preparing for a process improvement workshop could use the Process Improvement Copilot to gain an overview of process inefficiencies and generate initial improvement ideas. These ideas, along with their justifications, could then be used in the workshop to deal with the "blank-page" syndrome that psychologically hinders the generation of initial ideas (Cavallucci 2002) or to expand the list of ideas for discussion. Second, a person without BPI experience, such as a domain expert, could generate possible process improvement ideas. While domain experts regularly interact with the process, they usually know and can articulate process inefficiencies but may lack formal BPI knowledge. The Process Improvement Copilot could enable them to generate process improvement ideas and initiate further discovery of promising ideas with other stakeholders. Third, project managers overseeing large project transformation programs could employ the Process Improvement Copilot to scale new process models and their elements across the organization. For example, an organization that has developed blueprints for process improvement could upload these blueprints, along with detailed descriptions, into the knowledge base of the Process Improvement Copilot and apply these blueprints to all processes designated for review, thus substantially accelerating this review initiative.

Limitations and future research

The results of this research should be considered in light of several limitations, which also suggest avenues for future research. First, in its current version, the Process Improvement Copilot only addresses the BPI stages of inefficiency identification and idea generation. Other critical stages, such as idea evaluation or simulation of the to-be process design (Malinova et al. 2022), remain outside the current scope. Second, the Process Improvement Copilot is based on a limited amount of BPI knowledge and process context information, lacking a broader range of BPI knowledge (e.g., case studies with quantified impact of process improvement ideas) and process context (e.g., detailed business rules). Third, the underlying system architecture is implemented in its basic version, as the primary focus of the current research was to introduce a RAG-enhanced architecture for LLM-based PIISs and not to fine-tune the system architecture for a particular use case. A more robust technical evaluation of the individual RAG components was also omitted due to the primary focus on user-oriented design risks. Fourth, the perspective of computational complexity and costs for scaling the PIIS in a real-world context was not considered.

Beyond addressing the aforementioned limitations, several promising directions deserve further investigation. Future research could consider multi-technology PIISs that combine the potential of GenAI for idea generation with the strengths of other technologies for other BPI stages. The use of GenAI in PIISs can be explored further, potentially mapping out a comprehensive landscape of GenAI applications in process improvement. Moreover, enhancing the inefficiency identification stage with root-cause analysis capabilities could provide more nuanced insights into the origins of process inefficiencies, thereby enabling the generation of improvement ideas not for the observable consequences of the inefficiencies but for their root causes. Furthermore, incorporating reinforcement learning mechanisms through feedback loops for the generated process improvement ideas could enhance system performance in a collaborative human-AI environment. The RAG approach can also be advanced through the exploration of knowledge graphs (Xu et al. 2024) in the information retrieval step, the introduction of more complex components (Huang and Huang 2024), and effective orchestration of components (Gao et al. 2024). Finally, a comprehensive technical benchmarking of different RAG architectures for PIISs with a follow-up robust evaluation of the final system with additional focus on validity and replicability of the results (Gao et al. 2024; Salemi and Zamani 2024) would be a valuable initiative to estimate the scalability potential of RAG-enhanced LLM-based PIISs in real-world scenarios.

Appendix 1: Changes to the software prototype

Table 5 Changes to the software prototype suggested in the expert interviews

#	Change Description	Interviews	Priority	Status
1	Move navigation to sidebar and add clear page structure	6, 8, 11	must	done
2	Include examples for all manual input fields	11, 12	must	done
3	Justify each idea by showing the supporting piece of knowledge used for its generation	2, 8, 11	must	done
4	Expand output to include assumptions, implementation steps, and expected changes	4, 8	must	done
5	Create a user manual with key instructions	12	must	done
6	Integrate a step-by-step user guide within the prototype	2, 4, 8, 11, 12	should	covered with #5
7	Rank inefficiencies by significance (or provide data for human-on-the-loop to validate and rank them)	8, 11	should	done
8	Add option to regenerate improvement ideas	1	should	done
9	Display status message during task completion	1	should	done
10	Simplify the way of transferring inefficiency from table to the list of inefficiencies to solve	1	should	done
11	Separate process context and objectives into distinct, guided input fields	9, 12	should	done
12	Provide info on technical details of prototype (make backend transparent)	5, 6, 9	should	done
13	Add more public knowledge to the RAG knowledge base	3	could	ongoing
14	Expand the range of automatically recognized inefficiency patterns	5	could	ongoing
15	Offer alternative output formats (e.g., BPMN diagrams)	1, 10, 11, 13	could	future
16	Validate inefficiencies added manually to confirm their significance	1, 5	could	future
17	Collect user feedback (like/dislike + comment) on generated ideas (source of info for reinforcement learning)	1	could	future
18	Implement a conversational (chat) interface instead of single-message output	2, 3, 8	could	future
19	Allow inefficiency input via slide decks	3	could	future
20	Output ideas as a slide deck	3	could	future
21	Allow users to upload their own documents to the RAG knowledge base	6	could	future
22	Develop an automated pipeline for adding documents to RAG (e.g., papers from conferences)	6	could	future
23	Expand the amount of domain knowledge that can be uploaded	7	could	future
24	Expand pieces of knowledge from RAG with relevant case studies	3	could	future
25	Inform the user about the similarity between process inefficiency and the knowledge used	5	could	future
26	Suggest available or new resources (software, hardware, people, etc.) that could implement the idea	2	could	future
27	Shorten the output text	3	won't	-
28	Filter out irrelevant ideas	1	beyond the scope	future
29	Estimate the potential impact of solving each inefficiency on KPIs (alternatively, impact of each idea on KPIs)	1, 3, 4, 6, 7, 8, 11, 13	beyond the scope	future
30	Estimate the implementation complexity and time for each idea	3, 4, 8	beyond the scope	future

Appendix 2: Interview experts

Table 6 Company/Institution and research/practical experience of the interviewed experts

Interview	Expert	Company / Institution	BPM Experience	Years of Research Experience	Years of Practical Experience
Interview 1	Expert 1	Industrial IoT security company	Doctoral candidate; development of PM software	5	5
Interview 2	Expert 2	PM consulting company	BPM and PM projects across various industries	-	8
Interview 3	Expert 3	PM consulting company	Research papers; BPM and PM projects across various industries	1	7
Interview 4	Expert 4	Semiconductor manufacturer	Completed doctoral degree; PM projects in finance domain	5	6
Interview 5	Expert 5	Research institute	Doctoral candidate	3	-
Interview 6	Expert 6	Research institute	Doctoral candidate; BPM projects across various industries	5	4
Interview 7	Expert 7.1	Automotive OEM	PM projects	2	8
	Expert 7.2	Automotive OEM	PM projects	-	6
Interview 8	Expert 8.1	Aviation industry company	Completed doctoral degree; BPM projects	4	6
	Expert 8.2	Aviation industry company	BPM and PM projects	5	6
Interview 9	Expert 9	Research institute	Doctoral candidate; PM consulting projects	4	4
Interview 10	Expert 10	PM consulting company	BPI and PM projects across various industries	-	5.5
Interview 11	Expert 11	Research institute	Completed doctoral degree; BPM consulting projects	4	3.5
Interview 12	Expert 12.1	Research institute	Doctoral candidate; BPM consulting projects	2	3
	Expert 12.2	Research institute	Doctoral candidate; BPM consulting projects	2	2
Interview 13	Expert 13	Research institute	Research associate, BPM consulting project	2	1

Acknowledgements
This work was supported by the Bavarian Research Foundation (Bayerische Forschungsstiftung) [grant number AZ-1599-23].

Author contributions
Conceptualization: Uladzimir Smalei, Christoph Kecht; Funding acquisition: Wolfgang Kratsch, Maximilian Röglinger; Investigation: Uladzimir Smalei, Christoph Kecht; Methodology: Uladzimir Smalei, Christoph Kecht; Project administration: Christoph Kecht; Software: Uladzimir Smalei; Supervision: Wolfgang Kratsch, Maximilian Röglinger; Visualization: Uladzimir Smalei; Writing – original draft: Uladzimir Smalei, Christoph Kecht; Writing – review & editing: Uladzimir Smalei, Christoph Kecht; Wolfgang Kratsch, Maximilian Röglinger.

Funding
Open Access funding enabled and organized by Projekt DEAL. This work was supported by the Bavarian Research Foundation (Bayerische Forschungsstiftung) [AZ-1599-23].

Data availability
(1) The test event log used for demonstration and evaluation purposes is available from Celonis SE under the Celonis Academic Alliance license. The event log is available to all users with Celonis Academic Alliance license or from the authors upon reasonable request and with the permission of Celonis SE. (2) The real-world event log from a multinational technology conglomerate used for demonstration and evaluation purposes is not publicly available for confidentiality reasons. (3) The code of the software prototype is available in the GitHub repository: https://github.com/VladimirSmoley/process_improvement_copilot. (4) Recordings and transcripts of the interviews and the workshop are not publicly available for the confidentiality of the participants. The transcripts in the anonymized form are available from the authors upon reasonable request.

Declarations

Ethics approval and consent to participate
Not applicable.

Competing interests
The authors have no competing interests to declare that are relevant to the content of this article.

Received: 29 April 2025 / Accepted: 13 October 2025
Published online: 08 January 2026

References

Abbasi M, Nishat RI, Bond C, Graham-Knight JB, Lasserre P, Lucet Y, Najjaran H (2024) A review of AI and machine learning contribution in business process management (process enhancement and process improvement approaches). BPMJ. https://doi.org/10.1108/BPMJ-07-2024-0555

Ackermann L, Käppel M, Marcus L, Moder L, Dunzer S, Hornsteiner M et al (2024) Recent advances in data-driven business process management. Available online at https://arxiv.org/pdf/2406.01786

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. Inf. Syst. Front. 19(5):1101–1121. https://doi.org/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), Mexico City, Mexico. pp 3947–3960. https://doi.org/10.18653/v1/2024.naacl-long.219

Amershi S, Weld D, Vorvoreanu M, Fourney A, Nushi B, Collisson P et al (2019) Guidelines for human-AI interaction. In: Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems, vol 19. Association for Computing Machinery CHI, New York, NY, USA, 1–13. https://doi.org/10.1145/3290605.3300233

Bader M, Antony J, Jayaraman R, Swarnakar V, Goonetilleke RS, Maalouf M et al (2024) Why do process improvement projects fail in organizations? A review and future research agenda. IJLSS 15(3):664–690. https://doi.org/10.1108/IJLSS-07-2023-0126

Balaguer A, Benara V, de Freitas Cunha RL, de M. Estevão Filho R, Hendry T, Holstein D et al (2024) RAG vs fine-tuning: pipelines, tradeoffs, and a case study on agriculture. Available online at http://arxiv.org/pdf/2401.08406v3

Barredo Arrieta A, Díaz-Rodríguez N, Del Ser J, Bennetot A, Tabik S, Barbado A et al (2020) Explainable artificial intelligence (XAI): concepts, taxonomies, opportunities and challenges toward responsible AI. Inf Fusion 58:82–115. https://doi.org/10.1016/j.inffus.2019.12.012

Becker J, Kugeler M, Rosemann M (2012) Prozessmanagement. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-33844-1

Beerepoot I, Di Ciccio C, Reijers HA, Rinderle-Ma S, Bandara W, Burattin A et al (2023) The biggest business process management problems to solve before we die. Comput In Ind 146:103837. https://doi.org/10.1016/j.compind.2022.103837

Beerepoot I, van de Weerd I, Reijers HA (2019) Business process improvement activities: differences in organizational size, culture, and resources. In: Hildebrandt T, van Dongen BF, Röglinger M, Mendling J (eds) Business process management: 17th international conference, BPM 2019 Vienna, Austria. Springer, Cham, pp 402–418. https://doi.org/10.1007/978-3-030-26619-6_26

Beheshti A, Yang J, Sheng QZ, Benatallah B, Casati F, Dustdar S et al (2023) ProcessGPT: transforming business process management with generative artificial intelligence. In: 2023 IEEE International Conference on Web Services (ICWS), pp 731–739. https://doi.org/10.1109/ICWS60048.2023.00099

Bergener P, Delfmann P, Weiss B, Winkelmann A (2015) Detecting potential weaknesses in business processes. BPMJ 21(1):25–54. https://doi.org/10.1108/BPMJ-07-2013-0103

Buhl HU, Röglinger M, Stöckl S, Braunwarth KS (2011) Value orientation in process management. Bus Inf Syst Eng 3(3):163–172. https://doi.org/10.1007/s12599-011-0157-5

Cavallucci D (2002) TRIZ, the Altshullerian approach to solving innovation problems. In: Chakrabarti A (ed) Engineering design synthesis: understanding, approaches and tools. Springer, London, pp 131–149. https://doi.org/10.1007/978-1-4471-3717-7_9

Chen G, Shen Z, Iyer A, Ghumman UF, Tang S, Bi J et al (2020) Machine-learning-assisted de novo design of organic molecules and polymers: opportunities and challenges. Polymers 12(1). https://doi.org/10.3390/polym12010163

Chen H, Xiao Q, Yin X (2019) Generating music algorithm with deep convolutional generative adversarial networks. In: 2019 IEEE 2nd International Conference on Electronics Technology (ICET), pp 576–580. https://doi.org/10.1109/ELTECH.2019.8839521

Davis FD (1989) Perceived usefulness, perceived ease of use, and user acceptance of information technology. Mis Q 13(3):319. https://doi.org/10.2307/249008

Denagama Vitharanage IM, Bandara W, Syed R, Toman D (2020) An empirically supported conceptualisation of robotic process automation (RPA) benefits. In: Proceedings of the 28th European Conference on Information Systems (ECIS 2020), Online.

Dumas M, Rosa ML, Mendling J, Reijers HA (2018) Fundamentals of business process management. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-662-56509-4

Egger A, Fehrer T, Kratsch W, Wördehoff N, König F, Röglinger M (2025) Refining the process picture: unstructured data in object-centric process mining. Inf Syst 134:102582. https://doi.org/10.1016/j.is.2025.102582

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 (2023) Process-pattern.app: – A collection of business process redesign patterns. In: Fahland D, Jiménez Ramírez A, Kumar A, Mendling J, Pentland B, Rinderle-Ma S et al (eds) Proceedings of the Best Dissertation Award, Doctoral

Consortium, and Demonstration & Resources Forum at BPM 2023, co-located with 21st International Conference on Business Process Management (BPM 2023), pp 117–121.
Fehrer T, Fischer DA, Leemans SJJ, Röglinger M, Wynn MT (2022) An assisted approach to business process redesign. Decis Support Syst 156:113749. https://doi.org/10.1016/j.dss.2022.113749
Fehrer T, Marcus L, Röglinger M, Smalei U, Zetzsche F (2024) The FLAC method: Data-facilitated discovery of business process improvement options. In: Avital M, Karahanna E, Themistocleous M, Constantiou ID, Fitzgerald B, Seidel S (eds) Proceedings of the 32nd European Conference on Information Systems (ECIS 2024), Paphos, Cyprus.
Fehrer T, Moder L, Röglinger M (2025) A taxonomy for process improvement and innovation systems. Bus Inf Syst Eng. https://doi.org/10.1007/s12599-025-00928-4
Feuerriegel S, Hartmann J, Janiesch C, Zschech P (2024) Generative AI. Bus Inf Syst Eng 66(1):111–126. https://doi.org/10.1007/s12599-023-00834-7
Franzoi S, Hartl S, Grisold T, van der Aa H, vom Brocke J (2025) Explaining process dynamics: a process mining context taxonomy for sense-making. Process Science 2(1). https://doi.org/10.1007/s44311-025-00008-6
Gao Y, Xiong Y, Gao X, Jia K, Pan J, Bi Y et al (2024) Retrieval-augmented generation for large language models: a survey. Available online at http://arxiv.org/pdf/2312.10997v5
Groß S, Grisold T, Mendling J, Haase J (2024) Idea generation in exploitative and explorative business process redesign techniques. Inf Syst e-Bus Manage 22(3):527–555. https://doi.org/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 https://doi.org/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. BPMJ 27(8):25–56. https://doi.org/10.1108/BPMJ-03-2020-0116
Groß S, Yeshchenko A, Djurica D, Mendling J (2020) Process mining supported process redesign: matching problems with solutions. In: Serral Asensio E, Stirna J (eds) Proceedings of the Forum at Practice of Enterprise Modeling 2020, co-located with the 13th IFIP WG 8.1 Working Conference on the Practice of Enterprise Modeling (PoEM 2020), Riga, Latvia, pp 24–33.
Hammer M (2015) What is 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 3–16. https://doi.org/10.1007/978-3-642-45100-3_1
Han J, Kamber M, Pei J (2012) 2 - Getting to know your data. In: Han J, Kamber M, Pei J (eds) Data mining: concepts and techniques . Third edn. Morgan Kaufmann (The Morgan Kaufmann Series in Data Management Systems), Boston, pp 39–82. https://doi.org/10.1016/B978-0-12-381479-1.00002-2
Harl MV, Zilker S, Weinzierl S (2024) Towards automated business process redesign in runtime using generative machine learning. In: Avital M, Karahanna E, Themistocleous M, Constantiou ID, Fitzgerald B, Seidel S (eds) Proceedings of the 32nd European Conference on Information Systems (ECIS 2024), Paphos, Cyprus.
Heaven D (2019) Why deep-learning AIs are so easy to fool. Nature 574(7777):163–166. https://doi.org/10.1038/d41586-019-03013-5
Hofmann P, Rückel T, Urbach N (2021) Innovating with artificial intelligence: capturing the constructive functional capabilities of deep generative learning. In: Bui T (ed) Proceedings of the 54th Hawaii International Conference on System Sciences. HICSS 2021, Online, pp 5505–5514. https://doi.org/10.24251/HICSS.2021.669
Holtzman A, Buys J, Li D, Forbes M, Choi Y (2020) The curious case of neural text degeneration. Available online at http://arxiv.org/pdf/1904.09751v2
Huang L, Yu W, Ma W, Zhong W, Feng Z, Wang H et al (2025) A survey on hallucination in large language models: principles, taxonomy, challenges, and open questions. ACM Trans Inf Syst 43(2):1–55. https://doi.org/10.1145/3703155
Huang SY, Lee CH, Chiu AA, Yen DC (2015) How business process reengineering affects information technology investment and employee performance under different performance measurement. Inf Syst Front 17(5):1133–1144. https://doi.org/10.1007/s10796-014-9487-4
Huang Y, Huang J (2024) A survey on retrieval-augmented text generation for large language models. Available online at http://arxiv.org/pdf/2404.10981v2
Joshi A, Kale S, Chandel S, Pal DK (2015) Likert scale: explored and explained. BJAST 7(4):396–403. https://doi.org/10.9734/BJAST/2015/14975
Kaplan A, Haenlein M (2019) Siri, Siri, in my hand: Who's the fairest in the land? On the interpretations, illustrations, and implications of artificial intelligence. Bus. Horiz. 62(1):15–25. https://doi.org/10.1016/j.bushor.2018.08.004
Kecht C, Egger A, Kratsch W, Röglinger M (2023) Quantifying chatbots' ability to learn business processes. Inf Syst 113:102176. https://doi.org/10.1016/j.is.2023.102176
Kerpedzhiev GD, König UM, Röglinger M, Rosemann M (2021) An exploration into future business process management capabilities in view of digitalization. Bus Inf Syst Eng 63(2):83–96. https://doi.org/10.1007/s12599-020-00637-0
Kettinger WJ, Teng JTC, Guha S (1997) Business process change: a study of methodologies, techniques, and tools. Mis Q 21(1):55. https://doi.org/10.2307/249742
Khan A, Ghose A, Dam H, Syed A (2023) A survey of process-oriented data science and analytics for supporting business process management. Available online at http://arxiv.org/pdf/2301.10398v2
Kojima T, Gu SS, 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.
Korade NB, Salunke MB, Bhosle AA, Kumbharkar PB, Asalkar GG, Khedkar RG (2024) Strengthening sentence similarity identification through OpenAI embeddings and deep learning. IJACSA 15(4). https://doi.org/10.14569/IJACSA.2024.0150485
Koschmider A, Laue R, Fellmann M (2019) Business process model anti-patterns: a bibliography and taxonomy of published work. In: Proceedings of the 27th European Conference on Information Systems (ECIS 2019), Stockholm & Uppsala, Sweden.
Kourani H, Berti A, Schuster D, van der Aalst, WMP (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, Cham, pp 229–244. https://doi.org/10.1007/978-3-031-61007-3_18

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

Larrick RP (2004) Debiasing. In: Koehler DJ, Harvey N (eds) Blackwell Handbook of Judgment and Decision Making: John Wiley & Sons, Ltd. pp 316–338. https://doi.org/10.1016/j.dss.2020.113286

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.

Lewis P, Perez E, Piktus A, Petroni F, Karpukhin V, Goyal N et al (2020) 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), pp 9459–9474.

Li CY, Shinde T, He W, Lau SSF, Hiew MXB, Tam NTL et al (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. https://doi.org/10.1007/978-3-031-48424-7_21

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

Magaldi D, Berler M (2020) Semi-structured interviews. In: Zeigler-Hill V, Shackelford TK (eds) Encyclopedia of personality and individual differences. Springer International Publishing, Cham, pp 4825–4830. https://doi.org/10.1007/978-3-319-24612-3_857

Makarius EE, Mukherjee D, Fox JD, Fox AK (2020) Rising with the machines: a sociotechnical framework for bringing artificial intelligence into the organization. Journal of Business Research 120:262–273. https://doi.org/10.1016/j.jbusres.2020.07.045

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

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. https://doi.org/10.18653/v1/2020.acl-main.173

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

Mustansir A, Shahzad K, Malik MK (2022) Towards automatic business process redesign: an NLP based approach to extract redesign suggestions. Autom Softw Eng 29(1). https://doi.org/10.1007/s10515-021-00316-8

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 et al (eds) Enterprise, Business-Process and Information Systems Modeling. BPMDS EMMSAD 2011, London, UK: Springer. pp 88–102. https://doi.org/10.1007/978-3-642-21759-3_7

Nolle T, Luettgen S, Seeliger A, Mühlhäuser M (2022) BINet: multi-perspective business process anomaly classification. Inf Syst 103:101458. https://doi.org/10.1016/j.is.2019.101458

Oberdorf F, Schaschek M, Weinzierl S, Stein N, Matzner M, Flath CM (2023) Predictive end-to-end enterprise process network monitoring. Bus Inf Syst Eng 65(1):49–64. https://doi.org/10.1007/s12599-022-00778-4

Park G, van der Aalst, WMP (2022) Action-oriented process mining: bridging the gap between insights and actions. Prog Artif Intell. https://doi.org/10.1007/s13748-022-00281-7

Peffers K, Tuunanen T, Rothenberger MA, Chatterjee S (2007) A design science research methodology for information Systems research. Journal of Management Information Systems 24(3):45–77. https://doi.org/10.2753/MIS0742-1222240302

Radensky M, Shahid S, Fok R, Siangliulue P, Hope T, Weld DS (2025) Scideator: human-LLM scientific idea generation grounded in research-paper facet recombination. Available online at http://arxiv.org/pdf/2409.14634v3

Rai A, Constantinides P, Sarker S (2019) Editor's comments: Next-generation digital platforms: toward human-AI hybrids. MIS Q 43(1):3–9.

Raisch S, Krakowski S (2021) Artificial intelligence and management: the automation-augmentation paradox. AMR 46(1):192–210. https://doi.org/10.5465/amr.2018.0072

Reijers HA (2021) Business process management: the evolution of a discipline. Comput In Ind 126:103404. https://doi.org/10.1016/j.compind.2021.103404

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

Ribeiro J, Lima R, Eckhardt T, Paiva S (2021) Robotic process automation and artificial intelligence in industry 4.0 - a literature review. Procedia Comput Sci 181:51–58. https://doi.org/10.1016/j.procs.2021.01.104

Röglinger M, van Dun C, Fehrer T, Fischer DA, Moder L, Kratsch W (2021) Automated process (re-) design. In: Beerepoot I, Di Ciccio C, Marrella A, Reijers HA, Rinderle-Ma S, Weber B (eds) Proceedings of the International Workshop on BPM Problems to Solve Before We Die (PROBLEMS 2021), co-located with the 19th International Conference on Business Process Management (BPM 2021), Rome, Italy, pp 28–33.

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. https://doi.org/10.1007/978-3-642-45100-3_5

Salemi A, Zamani H (2024) Evaluating retrieval quality in retrieval-augmented generation. In: Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval, Washington DC, USA: Association for Computing Machinery, New York, NY, USA, pp 2395–2400. https://doi.org/10.1145/3626772.3657957

Schaschek M, Gwinner F, Neis N, Tomitza C, Zeiß C, Winkelmann A (2024) Managing next generation BP-x initiatives. Inf Syst e-Bus Manage 22(3):457–500. https://doi.org/10.1007/s10257-024-00681-3

Schuh G, Guetzlaff A, Schmitz S, Schopen M, Broehl F (2021) Event log-based weaknesses detection in business processes. In: 2021 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM), pp 734–738. https://doi.org/10.1109/IEEM50564.2021.9672975

Schulhoff S, Ilie M, Balepur N, Kahadze K, Liu A, Si C et al (2025) The prompt report: a systematic survey of prompting techniques. Available online at http://arxiv.org/pdf/2406.06608v6

Sellen A, Horvitz E (2024) The rise of the AI Co-pilot: lessons for design from aviation and beyond. Communications of the ACM 67(7):18–23. https://doi.org/10.1145/3637865

Shuster K, Poff S, Chen M, Kiela D, Weston J (2021) Retrieval augmentation reduces hallucination in conversation. In: Moens MF, Huang X, Specia L, Yih SWT (eds) Findings of the Association for Computational Linguistics: eMNLP 2021. Association for Computational Linguistics, Punta Cana, Dominican Republic , pp 3784–3803. https://doi.org/10.18653/v1/2021.findings-emnlp.320
Sonnenberg C, vom Brocke J (2012) 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. Springer, Berlin, Heidelberg. , pp 381–397. https://doi.org/10.1007/978-3-642-29863-9_28
Souza A, Azevedo LG, Santoro FM (2017) Automating the identification of opportunities for business process improvement patterns application. IJBPIM 8(4):252–272. https://doi.org/10.1504/IJBPIM.2017.088799
Stein Dani V, Leopold H, van der Werf JMEM, Beerepoot I, Reijers HA (2024) From process mining insights to process improvement: all talk and no action? In: Sellami M, Vidal ME, van Dongen B, Gaaloul W, Panetto H (eds) Cooperative Information Systems, Springer, Cham. https://doi.org/10.1007/978-3-031-46846-9_15
Tang J, Liu Y, Lin K-Y, Li L (2023) Process bottlenecks identification and its root cause analysis using fusion-based clustering and knowledge graph. Adv Eng Inf 55:101862. https://doi.org/10.1016/j.aei.2022.101862
Tiwari A, Vergidis K, Turner C (2010) Evolutionary multi-objective optimisation of business processes. In: Gao XZ, Gaspar-Cunha A, Köppen M, Schaefer G, Wang J (eds) Soft computing in industrial applications. Springer, Berlin, Heidelberg, pp 293–301. https://doi.org/10.1007/978-3-642-11282-9_31
van der Aalst, WMP (2013) Business process management: a comprehensive survey. International Scholarly Research Notes 507984. https://doi.org/10.1155/2013/507984
van der Aalst, WMP (2022) Process mining: a 360 degree overview. In: van der Aalst WMP, Carmona J (eds) Process mining handbook. Springer, Cham, pp 3–34. https://doi.org/10.1007/978-3-031-08848-3_1
van der Waal W, van de Weerd I, Beerepoot I, Lu X, Kappen T, Haitjema S et al (2025) Putting the SWORD to the test: finding workarounds with process mining. Bus Inf Syst Eng 67(2):171–190. https://doi.org/10.1007/s12599-023-00846-3
van Dun C, Moder L, Kratsch W, Röglinger M (2023) ProcessGAN: supporting the creation of business process improvement ideas through generative machine learning. Decis Support Syst 165:113880. https://doi.org/10.1016/j.dss.2022.113880
Vanwersch RJB, Shahzad K, Vanderfeesten I, Vanhaecht K, Grefen P, Pintelon L (2016) A Critical Evaluation and Framework of Business Process Improvement Methods. Bus Inf Syst Eng 58(1):43–53. https://doi.org/10.1007/s12599-015-0417-x
Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez A et al (2017) Attention is all you need. In: Guyon I, Von Luxburg U, Bengio S, Wallach H, Fergus R, Vishwanathan S et al (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. Eur J Inf Syst 25(1):77–89. https://doi.org/10.1057/ejis.2014.36
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. https://doi.org/10.1016/j.ijinfomgt.2015.10.002
Wastell DG, White P, Kawalek P (1994) A methodology for business process redesign: experiences and issues. The Journal of Strategic Information Systems 3(1):23–40. https://doi.org/10.1016/0963-8687(94)90004-3
Weinzierl S, Wolf V, Pauli T, Beverungen D, Matzner M (2022) Detecting workarounds in business processes - a deep learning method for analyzing event logs. Journal of Business Analytics 5(1):76–100. https://doi.org/10.1080/2573234X.2021.1978337
Willaert P, Van den Bergh J, Willems J, Deschoolmeester D (2007) The process-oriented organisation: a holistic view developing a framework for business process orientation maturity. In: Alonso G, Dadam P, Rosemann M (eds) Business process management. BPM 2007, Brisbane, Australia: Springer, Berlin, Heidelberg, pp 1–15. https://doi.org/10.1007/978-3-540-75183-0_1
Xu Z, Cruz MJ, Guevara M, Wang T, Deshpande M, Wang X et al (2024) Retrieval-augmented generation with knowledge graphs for customer Service question answering. In: Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval, Washington DC, USA: Association for Computing Machinery, New York, NY, USA, pp 2905–2909. https://doi.org/10.1145/3626772.3661370
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.
Zellner G (2011) A structured evaluation of business process improvement approaches. BPMJ 17(2):203–237. https://doi.org/10.1108/14637151111122329
Zellner G (2013) Towards a framework for identifying business process redesign patterns. BPMJ 19(4):600–623. https://doi.org/10.1108/BPMJ-Mar-2012-0020
Zemni MA, Mammar A, Hadj-Alouane NB (2016) An automated approach for merging business process fragments. Comput In Ind 82:104–118. https://doi.org/10.1016/j.compind.2016.05.002
Zuhaira B, Ahmad N (2021) Business process modeling, implementation, analysis, and management: the case of business process management tools. BPMJ 27(1):145–183. https://doi.org/10.1108/BPMJ-06-2018-0168

Publisher's Note