International Journal of Information Security (2026) 25:108
https://doi.org/10.1007/s10207-026-01245-x

REGULAR CONTRIBUTION

Context is key for cybersecurity: leveraging external knowledge for process model explanation via LLMs

Linda Maria Kölbel[1] · **Leo Poss**[1] · **Stefan Schönig**[1]

Received: 18 August 2025 / Accepted: 25 March 2026

Abstract

The gap between operational process design and the security regulation requirements represents a critical and underexplored source of cybersecurity risk. Business process models provide structured representations of system behavior but are routinely abstracted from external knowledge, including industry standards, organizational policies, and domain constraints, which are required to assess their security posture and verify regulatory compliance. To address this, we propose a *Security by Design* framework that leverages Large Language Models (LLMs) to systematically integrate structured process models with unstructured external knowledge for automated process explanation and compliance checking. Our approach combines BPMN process models with external security standards (ISO 27001 [International Organization for Standardization and International Electrotechnical Commission. [ISO/IEC 27001:2022] – Information security management systems – Requirements. ISO/IEC, Geneva, Switzerland, 2022. Fourth edition. Available from www.iso.org] and IEC 62443-3-3 [International Electrotechnical Commission. IEC 62443-3-3:2013 – Industrial communication networks – Network and system security – Part 3-3: System security requirements and security levels. IEC, Geneva, Switzerland, 2013. First edition. Available from www.iec.ch]) using a modular prompting architecture. We evaluate the framework using the LLM-as-a-Judge methodology on two real-world Industrial Internet of Things (IIoT) use cases, demonstrating accurate, contextually grounded results. We further introduce a four-part error typology to characterize model limitations in compliance-critical settings. While results are promising, human expert validation remains essential for nuanced regulatory interpretation. This work provides a methodological foundation for transparent, proactive cybersecurity by embedding context-aware compliance checks directly into the system design process.

Keywords Generative Process Intelligence · Large Language Models · Business Process Management · Process Models · Prompt Engineering

1 Introduction

The increasing interconnection of digital infrastructures, from smart cities and transportation networks to industrial control systems, has created a landscape of unprecedented complexity and vulnerability. As these systems expand, their attack surface grows, rendering traditional, reactive cybersecurity measures insufficient to counter sophisticated threats and targeted ransomware attacks. In this dynamic environment, a paradigm shift is necessary, moving from post-deployment security patching to a proactive approach grounded in the principle of *Security by Design* [3]. This principle mandates that security considerations be integral to the entire system design lifecycle, not an afterthought applied to a finished product [4].

A significant and often overlooked source of cybersecurity risk lies in the contextual gap between system design and regulatory requirements [5, 6]. Business process models, typically rendered in structured formats such as Business Process Model and Notation (BPMN), provide a clear description of the operational activities within a system. However, they are frequently abstracted from the vast body of external knowledge (including industry standards, regulatory frameworks, and organizational policies) that dictates why and how specific tasks must be performed to ensure security and compliance [7]. This challenge aligns with

✉ Linda Maria Kölbel
linda.koelbel@ur.de

Leo Poss
leo.poss@ur.de

Stefan Schönig
stefan.schoenig@ur.de

[1] University of Regensburg, Regensburg, Germany

Published online: 15 June 2026

emerging research on context-aware process intelligence, which emphasizes that effective process analysis requires not only processes in isolation but also comprehensive contextual information, including domain knowledge, event context, and organizational constraints [8, 9]. For instance, standards like the General Data Protection Regulation (GDPR) or the IEC 62443 [2] series for industrial cybersecurity specify detailed constraints, responsibilities, and conditions that are rarely embedded directly into the process models themselves, while standards such as ISO 27001 [1] provide the organizational and procedural foundation on which effective cybersecurity measures are built. This specific challenge of developing a process-oriented approach to IIoT security management has been the subject of ongoing research, highlighting the perspectives and challenges in this domain [10, 11]. This disconnect is not just an administrative inefficiency; it can be a direct source of vulnerability. The process of manually interpreting complex, text-based standards and applying them to operational models is a time-consuming, knowledge-intensive task and is critically prone to human misinterpretation or oversight. This gap between structured logic and unstructured rules substantiates the integrity of digital infrastructures and creates tangible weaknesses that threat actors can target. Existing literature highlights various applications of Natural Language Processing (NLP) to work with structured and unstructured data, ranging from process model construction and monitoring [12] to using NLP for vulnerability detection [13], as well as recent efforts to use Large Language Models (LLMs) for general cybersecurity [14] or across the Business Process Management (BPM) lifecycle (e.g., [15, 16]).

To address this challenge, this paper introduces a novel framework that operationalizes the *Security by Design* principle. By employing LLMs, we systematically connect structured process models with unstructured external knowledge sources [17, 18]. To evaluate the effectiveness of our approach, we apply the *LLM-as-a-Judge* methodology, a recognized evaluation framework where LLMs assess the quality and correctness of generated outputs, enabling systematic validation of compliance assessments [19, 20]. This leads to the central research question:

How can Large Language Models be leveraged to integrate external knowledge for the explanation and assessment of process model compliance against security standards?

To address this, our work contributes a validated Security by Design framework that operationalizes context-aware compliance checking. Furthermore, we provide a rigorous dual-evaluation of LLM efficacy in this domain, leading to a diagnostic error typology that shifts the automated compliance bottleneck from hallucination mitigation to the complex logical mapping of structured process semantics against unstructured regulatory rules.

2 Theoretical background

2.1 Large language models (LLM)

Large Language Models (LLMs) are a significant development in artificial intelligence, NLP, and the broader public. Incorporating NLP methods for managing unstructured data within BPM is already substantial [12], and the use of LLMs falls into a similar category. Using LLMs and their functionalities depends on their underlying architecture, which enables the model to determine the contextual importance of words, significantly enhancing language understanding [21]. Additionally, the combination of pre-training on vast text corpora and subsequent fine-tuning for specific tasks enables LLMs to generalize effectively across various domains [22]. Scaling up models by increasing the number of parameters has been shown to improve their ability to capture linguistic and semantic patterns from structured and unstructured data [22], while LLM performance remains highly sensitive to the choice of prompting strategies [23]. The absence of interpretability and traceability makes it difficult to understand the rationale behind outputs [23, 24]. Furthermore, they can reproduce and exacerbate biases present in training data, and their expressive capabilities remain limited for certain applications [23, 24]. In addition, LLMs are prone to hallucinations and produce information that appears plausible but might be incorrect [23]. Our approach relies on using LLMs as the black box they are, enriching them with external context, and enhancing prompt results by leveraging both implicit knowledge encoded within the LLM and external context to simplify explanations and comparisons between different parameters for the user.

2.2 Business process management (BPM)

Business Process Management (BPM) encompasses systematic approaches to enhance organizational efficiency and effectiveness through process optimization, cost reduction, and operational acceleration [25]. Beyond operational improvements, BPM provides comprehensive decision support mechanisms that enable organizations to maintain a competitive advantage through enhanced process control and strategic management capabilities [26, 27]. Contemporary BPM frameworks integrate methodological, technical, and analytical components to systematically identify, discover, analyze, redesign, execute, and monitor business processes while optimizing performance outcomes [26]. BPM operates through iterative lifecycle frameworks comprising distinct phases that facilitate continuous improvement through cyclical execution [28]. The lifecycle's architectural design varies across theoretical frameworks, with different scholarly approaches proposing alternative phase configurations and

descriptive methodologies, reflecting the discipline's evolving conceptual foundations and practical applications.

Business processes are defined as a collection of interrelated events, activities, and decision points involving several actors and objects, which collectively lead to an outcome of value to at least one customer [26]. A business process aims to create value for customers or other stakeholders by efficiently and transparently producing a desired result; as such, it structures and standardizes workflows within organizations. They help to increase efficiency and consistency, assign responsibilities, monitor the quality of products and services, meet compliance requirements, and analyze and improve process flows [26]. Visual representations of them are suitable for understanding and modeling processes [29] and can be provided in a structured, machine-readable format for automated execution by process engines. In addition to these structured formats, various modeling languages provide mechanisms for extending standard features with functionality tailored to specific domains and needs, such as the Internet of Things (IoT) [30], location [31], and healthcare [32]. Processes are not always explicitly modeled in a structured way; they can also exist as unstructured, textual descriptions. This connection between textual descriptions and process modeling is shown in [12] (see Section 4) and further motivates our research.

3 Research methodology

This research employs the Design Science Research (DSR) methodology [33] to systematically investigate the integration of LLMs with process models using external context. DSR provides a scientifically rigorous framework for creating artifacts that address practical problems while generating theoretical insights [34]. We follow the five-phase DSR method [35]:

1. **Problem Identification and Motivation**: We identify the need for a structured representation of process models that can respond to external context queries and evaluate conformity against external conditions. Current research gaps are analyzed through a comprehensive literature review.
2. **Artifact Design and Development**: Our approach integrates appropriate technologies with a novel prompting strategy that synthesizes process models and external context information into comprehensive LLM prompts. The artifact undergoes iterative refinement based on preliminary testing and literature insights. In the first iteration, an initial framework was developed based on the collected literature. The selection and formulation of the prompt were then improved through several test runs. The questionnaire used was also refined in several rounds to improve the impact.
3. **Demonstration**: We demonstrate how we can collect relevant information about compliance with the controls of ISO 27001 using an information security process by contributing specific contextual information. In a second use case, we demonstrate the artifact using an existing BPMN process model enriched with IEC 62443-3 security features [36], highlighting compliance analysis capabilities against this industrial standard.
4. **Evaluation**: The artifact's performance is assessed through a predefined catalog of questions and answers that cover both model comprehension and compliance verification, providing insights into its strengths and limitations. For this, we adapted the previously mentioned LLM-as-a-Judge paradigm and, for the second use case, an additional layer of manual evaluation, including statistical measures, and comparing the reliability of both approaches for use with BPM.
5. **Communication**: Research findings, including design processes and evaluation outcomes, are documented to ensure transparency and reproducibility.

The iterative foundations of this framework provide a structured approach for developing and improving the artifact. They are directly visible in the two-part evaluation, split into an initial phase for testing the functional correctness of the approach and a second phase in which we check and quantify our results based on previous versions (e.g., iterative improvement of prompts and prompt combinations).

4 Related work

4.1 BPM and NLP

The authors of [12] have already shown that process models are closely related to their textual description. Within the overall research endeavor to investigate how NLP can improve BPM, they define three levels: multi-process management, single-process management, and process instance management. Within this broad field, they also discuss the challenges of NLP in BPM, including semantic variability (process descriptions in natural language can be ambiguous, and domain-specific terms may not always be interpreted correctly by NLP) and the inherent complexity of integrating NLP into existing BPM systems. Additionally, the application of NLP in BPM enables automated process modeling from textual process descriptions and enhances capabilities for process analysis and real-time monitoring. All these functionalities and approaches are closely related to the basic idea behind LLMs and transformer-based models: discovering latent (hidden) information in large, often unstructured data within BPM.

Similar appliances, as presented below, can be found in [37], which investigate the efficacy of LLMs for process information extraction from natural language texts, addressing inherent linguistic ambiguities and semantic processing challenges, [38] which propose algorithmic frameworks for automatically deriving business process models from organizational documentation, employing advanced NLP techniques to identify process elements and control flow dependencies and the systematic literature review by [39] that establishes foundational taxonomies for NLP-BPM convergence, identifying critical research gaps in tool integration and semantic model validation. Unlike the aforementioned research fields, we do not use LLMs to translate textual descriptions into process models; instead, we leverage NLP via LLMs to combine structured and unstructured data (i.e., by leveraging process models and model-specific information to generate textual descriptions, explanations, and reviews of the models).

4.2 BPM and LLMs

This leads to the main focus of the related work, in which we review the literature on the benefits of using LLMs in BPM that incorporate external contextual information. To achieve this, we conducted a systematic literature review, following the proposed methodology by Okoli [40]. A detailed search strategy was developed to ensure a comprehensive and systematic identification of relevant work. This process involved formulating precise search terms and keywords aligned with the research questions, constructing structured search strings, selecting appropriate academic databases, and defining practical screening criteria to filter the results effectively. The databases selected for the literature search include *SpringerLink*, *ScienceDirect*, *IEEE Xplore*, and the *ACM Digital Library*. These databases were selected for their strong focus on computer science and information systems, providing comprehensive coverage of the core literature on both BPM and LLMs. Based on the research question and derived keywords, we applied the following search string: ("Business Process*" *OR* BPM *OR* "Business Process Model" *OR* "Business Process Management") *AND* ("Large Language Model" *OR* LLM *OR* "Generative AI" *OR* "Generative Artificial Intelligence").

To ensure the relevance and quality of the identified literature, a multi-stage screening process with a practical set of inclusion and exclusion criteria was applied. These criteria were developed to retain only studies that directly addressed the formulated research questions concerning the application of LLMs in BPM. Initially, all 3,308 identified records were screened for their titles and abstracts. The primary inclusion criterion for this stage was a clear, direct focus on the intersection of LLMs and BPM. Following this, the remaining 220 papers were assessed for full-text eligibility, where more specific criteria were applied: Inclusion Criteria **(IC1)** *Direct Application in BPM*: The core of the research had to feature a substantive application of an LLM for a specific task within the BPM domain, **(IC2)** *Mapping to BPM Lifecycle*: The study needed to apply an LLM in a way that could be clearly attributed to at least one phase of the BPM lifecycle [26], such as process identification, modeling, analysis, or redesign. Exclusion Criteria **(EC1)** *Content not Relevant*: Papers that do not address topics relevant to the presented work in terms of content, but focus on a different scientific contribution. **(EC2)** *Marginal Topic*: Papers were excluded if the connection between BPM and LLMs was treated as a peripheral topic or only mentioned in passing, rather than being the central focus of the research. This includes, for example, the work of [41], which examines the potential advantages of integrating generative artificial intelligence into Industry 4.0/5.0 and, while addressing industrial processes, only discusses the connection to LLMs and security in a rudimentary way. **(EC3)** *Lack of Direct Application*: Studies that did not demonstrate a concrete application or integration of LLMs with BPM were discarded. This included theoretical papers or those that suggested future possibilities without a proof of concept or implementation. One example is the work of [42], which discusses the potential integration of GenAI into BPM, focusing on process optimization, automation, and decision support at a theoretical level, without implementation or execution.[1]

Applying these screening criteria resulted in the exclusion of 220 reports, narrowing the initial findings to a thematically relevant set of 20 studies. This methodical approach ensured that the final literature review was grounded in a solid foundation of papers that directly investigate the application of LLMs across the BPM lifecycle. The PRISMA diagram in Figure 1 illustrates the application of individual screening to the results. The articles identified are shown in Table 1. These were divided into the phases of the BPM lifecycle in which LLM supports BPM.

Throughout the BPM lifecycle, we aim to focus on the following selection of publications for further insights. In the process identification phase, [44] proposes using LLMs as co-pilots to classify unstructured documents in BPM. They apply targeted prompting and iterative refinement to enhance classification performance. While their approach shows potential, initial results reveal limitations in precision and recall. The authors suggest further work to address prediction errors and improve accuracy, highlighting the need for optimization before LLMs can be reliably used for document classification in BPM. In [16], the authors integrate LLMs into process modeling by developing an interactive prototype that enables experts to describe processes using

[1] A list of the intermediate and final publications identified in the SLR can be found in the supplementary material.

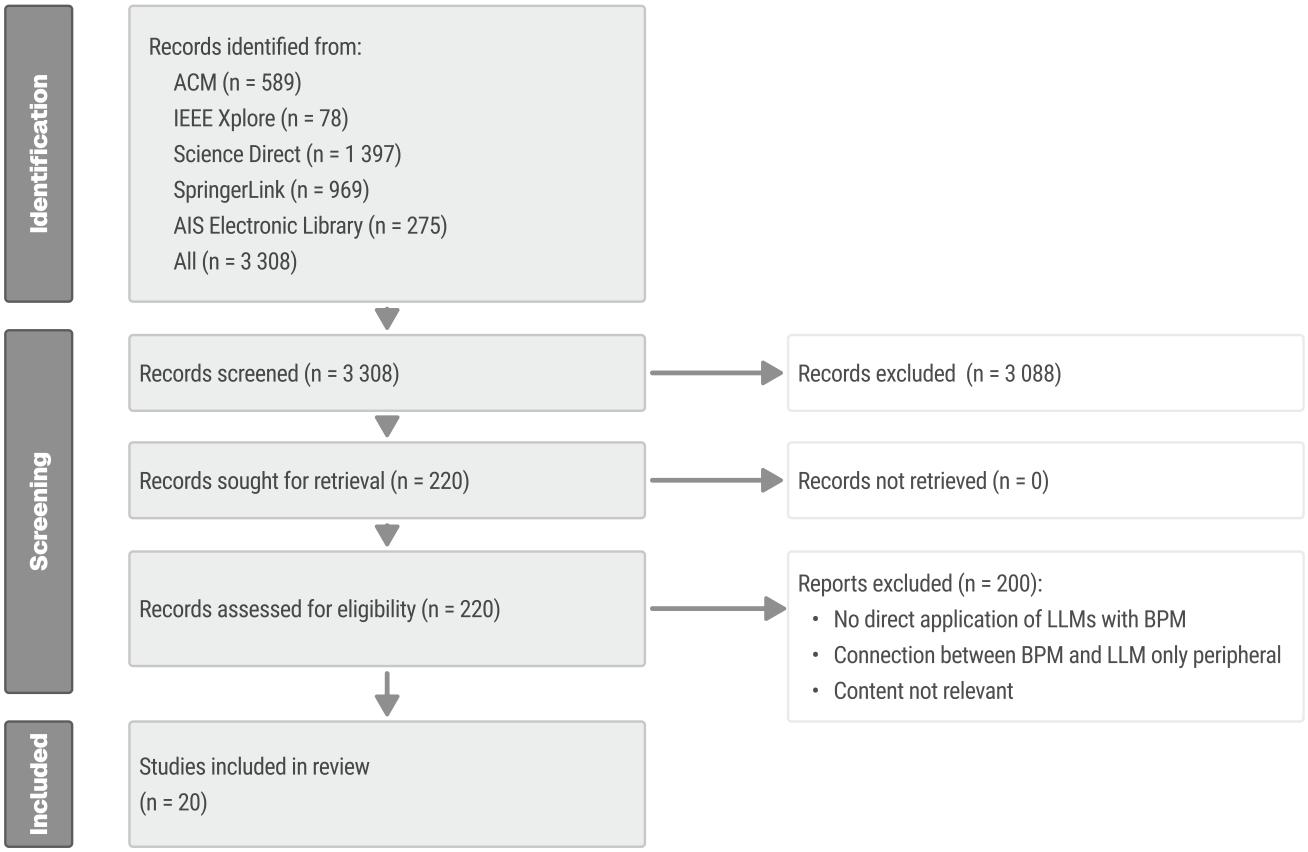

Fig. 1 PRISMA diagram [43] of structured literature review

natural language. The system they developed translates these descriptions into a domain-specific language and provides immediate formal feedback. Unlike our work, an LLM is used here to create a model rather than to explain it; the authors also do not consider external context. In [45], the authors examine how LLMs can be used to identify and optimize waiting times in business processes during the process analysis phase. They develop a prompting method that enables an LLM to analyze the specific causes of waiting times based on event logs, suggesting ways to reduce them. The study compares minimalist prompting strategies with an extended method that incorporates detailed redesign patterns. Considering prompting methods is particularly relevant to our work. For redesign and implementation, [46] introduces a generative process orchestration approach that utilizes pre-trained LLMs to analyze historical automation data and generate optimized, adaptive process flows. Their method dynamically integrates software components in line with best practices. Despite promising results, the authors note limitations related to data quality and stress the importance of human validation.

4.3 Compliance checking in cybersecurity

Beyond process modeling and redesign, compliance checking represents a critical area where LLMs are being applied to automate risk assessment and policy verification. In cybersecurity, compliance checking presents unique challenges due to the technical precision required, the rapidly evolving nature of security policies, and the need to integrate both structured configurations and unstructured documentation.

Recent work demonstrates the growing use of LLMs for compliance detection and verification. The authors of [47] introduce a framework that leverages LLMs to interpret security specifications and detect discrepancies between standard and actual business processes. Their approach demonstrates the efficacy in identifying compliance risk points by analyzing textual policy documents. In the context of GDPR compliance, the approach of [48] integrates LLMs with a Complex Event Processing engine to enable real-time detection of security policy violations while providing explanations for identified breaches. This work highlights the potential of LLMs not only to detect violations but also to generate human-readable justifications. Furthermore, in the regulatory technologies domain, LLMs are employed

for continuous compliance monitoring by comparing system states against acceptable regulatory values [49].

While these approaches demonstrate the utility of LLMs for compliance detection, checking, and verification, a critical gap remains in the systematic integration of structured and unstructured data. Established methods for security compliance checking, such as formal verification, model checking, and policy-based frameworks like Security-by-Contract, rely primarily on structured models and configurations [15]. Conversely, LLM-based compliance work has focused predominantly on text-to-model generation or on generic Retrieval-Augmented Generation (RAG) approaches that operate on unstructured documentation. The challenge of systematically combining structured security configurations with unstructured contextual information, such as technical documentation, organizational policies, and domain knowledge, remains largely unaddressed. This integration is particularly crucial in cybersecurity compliance, where security policies must be validated against both technical system configurations and contextual business requirements.

Based on our systematic review, research has demonstrated the extensive utility of LLMs across the BPM lifecycle, with primary applications in process identification, modeling, and analysis. Recent work has extended this to compliance checking and verification, particularly in cybersecurity contexts, as discussed in the previous subsection. However, despite these advances and the existence of established, structured methods for security compliance verification, such as formal methods, model checking, and Security-by-Contract frameworks, a critical gap persists. Current LLM-BPM research largely fails to systematically integrate structured process models with complex, unstructured external contexts, such as technical documentation, organizational policies, and domain-specific knowledge bases. This limitation is particularly problematic for modern compliance audits, which require reasoning across heterogeneous data sources to validate both technical configurations and contextual business requirements. This article directly addresses this gap by developing a methodology that systematically integrates structured process models with unstructured external knowledge (e.g., regulatory documents, organizational policies, and domain-specific standards) using LLMs and advanced prompting strategies, enabling automated, context-aware compliance verification and explanation of business process models within cybersecurity and regulatory contexts.

5 Artifact design and development

5.1 Concept

Motivated by the research question of how external information can be used to explain and contextualize process models, this section defines the concrete artifact developed. It situates it within the broader research landscape encompassing NLP, BPM, and the integration of LLMs. We describe the following design objectives for DSR [34]: *(i)* The artifact needs to be able to accept different types of process models and modeling languages in a structured representation, *(ii)* contexts from various domains and dimensions must be incorporable, and *(iii)* relying on best practices for the use of LLMs, we apply different prompting strategies and compare them against each other. The artifact shown in Figure 2 gives an overview of our concept and the main features: With the LLM in the center, it also contains the *inputs*, the *outputs*, and *settings for querying the LLM* in the form of prompting strategies.

5.1.1 Process

An essential part of process explanation and BPM is the process itself, which is provided in a form that can be processed by an LLM. To enable context-aware analysis, the process model must support attaching references to external information. For example, when assessing compliance with a corporate policy, the process model must incorporate elements relevant to that policy; the design and structure of the process model depend on the context in which it is applied.

5.1.2 Context

An external context, as applied in process modeling and LLM-based reasoning, is a multi-dimensional construct that provides essential background and situational information for the comprehensive understanding, explanation, and verification of process models. This context is typically available in a non-structured, often textual format and can be systematically categorized as follows:

Process Model Context. This dimension comprises information directly derived from or descriptive of the process itself. It includes defined process objectives, data objects manipulated within the process, historical data from previous executions, any BPMN extensions (e.g., for security or IoT), and documented process variants. Such structured context is crucial for process-aware information systems, as it supports the comprehension and analysis of process models [26].

LLM Context Methods. This category encompasses the various techniques that enable LLMs to process and utilize information from the process model and broader domain context. These methods include prompt engineering, strategies for integrating diverse context sources, the use of few-shot examples, the generation and application of semantic embeddings, and techniques such as RAG to dynamically fetch and incorporate relevant external knowledge. These approaches are essential for tailoring the LLM's reasoning and ensuring responses are contextually grounded [62].

Table 1 Overview of the articles and their addressed BPM phases

Reference	BPM Lifecycle Phase						Key Contribution	Limitations
	Identi- fication	Modeling	Analysis	Redesign	Implementation	Monitoring		
Beheshti et al. [50]	○	○	●	○	○	●	ProcessGPT uses LLMs to improve data- and knowledge-heavy processes with a context-aware decision support system.	Requires high-quality data and struggles with integration in changing business environments.
Bernardi et al. [15]	○	○	●	○	○	●	Proposes a framework combining RAG and fine-tuning for context-aware decision support in BPM.	Depends on the quality of data chunking and is complex to adapt to different process models.
Cobo-Ariza et al. [48]	○	●	●	○	○	●	Architecture integrating BPMS, security policy modeling, and Complex Event Processing engine to detect security policy violations in real-time and use LLMs to explain GDPR compliance.	Solely for GDPR and specific extension, focuses on compliance, not explicitly on process specifics.
Estrada-Torres et al. [51]	○	●	●	○	○	○	Conducts a systematic literature review on integrating LLMs across BPM phases.	Notes limited reproducibility of results and a lack of standard evaluation methods.
Grohs et al. [17]	○	●	○	○	○	○	Generates imperative and declarative process models from natural language descriptions.	The model's output can be inconsistent for similar inputs due to its non-deterministic nature.
Ilagan et al. [44]	●	○	○	○	○	○	Proposes using LLMs as co-pilots for automating document classification in BPM.	The model did not perform as well as expected in document classification.
Kampik et al. [18]	○	●	●	●	○	○	Combines fine-tuned LLMs with BPM tools for context-specific modeling, analysis, and improvement suggestions.	Faces challenges with data management, reliability, and the need for human oversight.
Kogler et al. [16]	○	●	○	○	○	○	Uses LLMs to help domain experts model processes in natural language, converting them into a formal JSON format.	Lacks global context and has limited UI elements, impacting usability and modeling accuracy.
Kourani et al. [52]	○	●	○	○	○	○	Presents a framework using LLMs and an intermediate representation to generate executable process models iteratively from text.	Needs to include more process details, conduct wider testing, and improve user interaction.
Lashkevich et al. [45]	○	○	●	●	○	○	A prompting method allowing LLMs to analyze event logs and suggest specific redesigns.	Output quality varies depending on how specific the prompt is, and it's not easily applied to many different process situations.

Table 1 continued

Reference	BPM Lifecycle Phase						Key Contribution	Limitations
	Identi- fication	Modeling	Analysis	Redesign	Implementation	Monitoring		
Licardo et al. [53]	○	●	○	○	○	○	Proposes a method combining NLP, deep learning, and LLMs to get accurate BPMN models from text automatically.	Relies on the quality of the input text and doesn't fully support certain BPMN elements.
Minor and Kaucher [54]	○	●	●	○	●	○	Develops a RAG system using LLMs to create text explanations for BPMN process models automatically.	Explanation quality drops if the text exceeds 3800 tokens, showing the model's capacity limit.
Pasquadibisceglie et al. [55]	○	○	●	○	○	●	Transforms event log data into semantic text histories and uses a BERT model to predict activity suffixes with better accuracy.	Depends on high-quality event log data and sometimes faces difficulty interpreting the model's predictions.
Paulose and Neelanath [46]	○	○	○	●	●	○	Introduces Generative Process Orchestration, using LLMs to analyze past workflows and create optimized process designs and implementations dynamically.	Relies on data quality and needs human checking to be reliable.
Schnepf et al. [56]	○	○	○	●	●	○	Explores using LLMs for automation in ERP systems, showing they can run and optimize complex processes on their own.	Requires tuning for specific domains and struggles to ensure consistent execution across different process types.
Softic et al. [57]	○	○	●	●	○	○	Uses LLMs to analyze employee feedback sentiment to help with process redesign and making informed decisions.	Depends on the quality and clarity of comments, and may have biases in interpreting sentiment.
Toslali et al. [58]	○	○	●	○	●	○	Uses LLMs to automate and speed up Third-Party security risk management, greatly reducing the time needed.	Relies on high-quality document data and might face issues handling complex or unclear security assessments.
Vidgof et al. [59]	●	●	●	●	●	●	Systematically examines how LLMs are used across the entire BPM lifecycle, identifying applications from finding processes to monitoring them.	Focuses on the relationship between LLMs and BPM without considering outside factors.
Ziche and Apruzzese [60]	○	●	○	○	○	○	An LLM-based chatbot that simplifies process modeling using RAG and BPMN Sketch Miner.	Challenges include customizing models for specific organizations and ensuring accurate modeling.
Barrientos et al. [61]	○	○	●	○	○	○	Automated hybrid approach to analyze the impact of regulatory requirement changes on process compliance.	Precision affected by representational inconsistencies in formalization and redundancy.

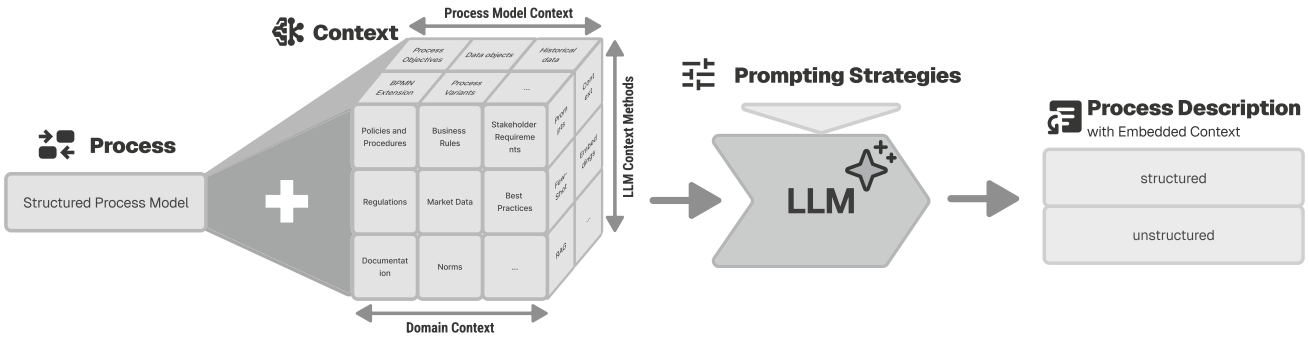

Fig. 2 Framework for using external knowledge with LLMs for process explanation

Domain Context. This dimension refers to the broader external knowledge and situational information relevant to the process's operational environment. It encompasses organizational policies, business rules, stakeholder requirements, regulatory frameworks (e.g., IEC 62443), market data, best practices, and established norms, as well as strictly formally defined data in the form of ontologies. This broader context is essential for ensuring that process models align with external requirements and constraints, as well as for supporting compliance, stakeholder alignment, and adaptability in dynamic environments. The wide array of possibilities provides an initial set of domain context information, but is not limited to these examples, leveraging the generalization capabilities of transformer-based language models.

5.1.3 LLM

The LLM needs to identify relevant information in the process model and map it to the context. Explicit context that is transferred to the LLM is the basis. The LLM contains implicit, trained additional context, which is also used to understand and explain the process models concerning context-related questions. LLMs match and connect process descriptions and external context.

Various external contexts and process models can be selected as input and transferred to the chosen LLM using a suitable prompting strategy. The LLM acts as a black box that outputs a structured or unstructured process description, considering the information requested in the prompt about the external context. For the result, we differentiate between structured and unstructured content: An unstructured process description is a verbal explanation of the process model and its elements, while a structured process description returns binary or constrained responses on the process model's compliance.[2]

5.1.4 Prompting strategies

To achieve the best possible results, a prompting strategy that incorporates the process model, the external context, and a specific question must be used to pass this information to the LLM. The strategy for prompting is the setting in which the LLM formulates precise objectives. Based on a review of existing literature and current best practices, we identified suitable strategies, which lead to the following state-of-the-art prompting strategies and baseline [63][3] and can be found in the linked repository:

- **Zero-shot Prompting:** This baseline strategy involves posing a direct question to the model without providing any examples or supplementary context. It relies entirely on the model's pre-existing knowledge to generate a response.
  - *Example: "Answer the following question: [Question]"*
- **Few-shot Prompting:** This strategy includes one or more illustrative examples within the prompt to guide the model's response. These examples demonstrate the desired format, style, and logic for the answer.
  - *Example: "Orient yourself with the following examples: Question: [Question] Answers: [ExampleAnswer1], [ExampleAnswer2] [...]"*
- **Chain-of-Thought (CoT) Prompting:** This approach instructs the model to break down its reasoning process into explicit, sequential steps before delivering the final answer. By specifying a procedure in the prompt, this strategy aims to improve logical coherence and completeness, which is particularly useful for complex analysis and evaluation tasks.

[2] We explicitly list both options, as just going for a binary decision with LLMs might not work, but trimming down numerous pages of process documentation and context into a concise description can also provide aid to human actors.

[3] A comprehensive list of identified literature is available from the authors upon request. Examples are excerpts directly taken from both use cases below.

  - *Example: "Analysis Approach: 1. Understand: Identify the key security aspects in the question; 2. Locate: Find relevant elements in the BPMN model; 3. Analyze: Check for security controls and compliance with IEC 62443-3-3; 4. Conclude: Provide a clear, evidence-based response."*

- **Role Assignment Prompting:** This strategy directs the model to adopt a specific persona or expertise, thereby shaping the tone, scope, and depth of its response. Answering from a predefined point of view enhances the relevance of the output in specialized domains.
  - *Example: "You are an expert in Information Security and ISO 27001 compliance. Analyze the provided BPMN process model and answer questions about its security aspects and compliance with ISO 27001 requirements and Annex A controls."*

- **Context Integration Prompting:** This method involves embedding additional background information, such as standards, policies, or process-specific knowledge, directly into the prompt. This enables the model to base its reasoning on relevant, up-to-date, domain-specific content, thereby enhancing its performance on tasks that require specialized knowledge.
  - *Example: "SIREN (Security IoT process Notation) is a BPMN-based approach for modeling and monitoring security-aware Industrial Internet of Things (IIoT) processes, aligning with the IEC62443 standard. [...] "*

- **Format Prompting:** This technique instructs the model to structure its output in a specific machine-readable format, such as JSON. This is essential for ensuring the response can be programmatically parsed and used in downstream applications.
  - *Example: "Response Format Guidelines: 1. Role & Tone: Act as a precise information security and compliance analyst. Be direct, objective, and concise in your response. 2. Scoping and Data Source: [...]"*

5.2 Implementation

To operationalize our conceptual framework, we developed a modular pipeline in Python, designed for extensibility and reproducibility. The implementation's architecture is centered around a main script that orchestrates data loading, prompt generation, interaction with various LLMs[4], and the following evaluation.[5] The complete source code, experimental data, and Jupyter Notebooks used for analysis are available in the project repository for transparency and replication purposes.

The general workflow of our implementation is implemented generically and not tied to the evaluation use case. It proceeds as follows:

1. **Data Preparation**: The process begins by loading and preprocessing the necessary artifacts. The external context (here: the IEC 62443-3-3 standard) is extracted from a Markdown file that includes the complete standard, structured into a data frame by the document structure and the different requirements and SLs. Similarly, the catalog of 71 evaluation questions is loaded from an Excel file, which includes the questions, categories, and sample answers for evaluation and few-shot prompting. For single use, we can directly provide few-shot examples and context to use for the prompt.
2. **Prompt Generation**: A core component is a dynamic `generate_prompt` function that constructs the final prompt string by selectively combining different components based on a configuration object. It assembles the prompt by integrating the question, the process model (read from its annotated .bpmn XML file), and the relevant segment of the external context, along with optional elements like role instructions, few-shot examples, and chain-of-thought steps. This modular design allowed us to systematically test the various prompting strategies and remains adaptable for future changes.
3. **LLM Interaction**: We implemented a generic `LLMWrapper` class to handle communication with different LLM providers and models. This wrapper provides a unified interface for sending prompts and receiving responses from various endpoints, including Google's *Gemini API*, *OpenRouter*, and local models served via *LM-Studio*. The execution of queries across the entire question catalog for each prompt configuration was parallelized using a `ThreadPoolExecutor` to improve efficiency.
4. **Automated Evaluation (LLM-as-a-Judge** [19, 20]): To complement our manual analysis, we implemented an automated assessment pipeline using an *LLM-as-a-Judge* approach. This method uses a separate LLM to provide a systematic evaluation of the primary model's answers against the ground-truth samples. The evaluator LLM is guided by a detailed prompt that defines specific criteria for a set of evaluation statuses (e.g., "Fully Correct," "Partially Correct") and requires the output to be in a strict JSON format containing an analysis, an evaluation status,

[4] https://openrouter.ai/, https://aistudio.google.com/, https://lmstudio.ai/, https://ollama.com/

[5] The repository for the implementation can be found at https://github.com/LeoPoss/contextIsKey.

and key findings. This structured process was designed to enforce a consistent evaluation framework across all 355 model-generated answers. The final, comprehensive results were compiled into a single data frame for analysis and visualization.

6 Evaluation

Our evaluation framework adopts established DSR methodology, specifically incorporating the Framework for Evaluation in Design Science Research (FEDS) [64]. This approach provides methodological rigor through a balanced assessment of both artificial and naturalistic evaluation contexts. Following the recommendations for continuous evaluation throughout the DSR process [65], we implemented an evaluation strategy corresponding to the 'Human Risk & Effectiveness' evaluation pathway. The first use case comes from a service company and addresses compliance with the ISO 27001 standard based on the process for responding to information security incidents. This process comes from the company's process map and is modeled in BPMN. The process XML, its process description, a list of all company policies, and the requirements of the ISO 27001 standard are used as context to verify compliance with the ISO 27001 controls. The second use case involves a production process of a manufacturing company that must be verified for compliance with the IEC 62443-3 requirements. For this purpose, security-related information was added to the process model using a suitable BPMN extension. This extended process model is checked for compliance with the controls of IEC 62443-3-3, where the requirements of IEC 62443-3 and a description of the extension notation were used as external context. The functional evaluation assesses whether our concept produces useful results as specified in [34], and includes technical feasibility assessment [66] and correctness verification [67]. We first verify the technical correctness and feasibility of the concept based on its implementation, and then assess the results of the experiments based on the factual correctness of the questions for the exemplary process models. Similar to [68], which verifies the compliance of process models with the GDPR, we apply two distinct real-world processes to evaluate the usability of our approach in relation to two norms.

6.1 Using the ISO 27001 for security management

For our first use case, we utilize a real-world process model employed by a software provider to address its security issues.[6] To achieve this, we aim to verify the compliance of the incident response process, which outlines how security should be managed within an organization, in accordance with ISO 27001. To achieve compliance with this standard, we aim to determine whether the company's incident response process fulfills the controls outlined in ISO 27001. Currently, this manual comparison of the requirements of a standard with the information in a comprehensive, process-dependent manual and process documentation, as well as checking process conformance and compliance[7], is a time-consuming process that requires a deep understanding of both the business processes and the corresponding standards (as seen in an ongoing research project and the more general problem of structuring, processing, and using large amounts of unstructured data). Since standards, process models, and questions used in our framework are interchangeable and can be selected individually for each organization, according to its needs, this reduces the amount of expert and domain knowledge required, as well as the time needed to verify compliance with standards. A recent research project showed that manually comparing the requirements of a standard with the information in a comprehensive, process-dependent manual takes significantly longer. The developed framework is intended to reduce this time investment.

6.1.1 Structure

As our first use case, we will utilize the incident response process of a software provider company, as shown in Figure 3. The Information Security Incident Management Process begins with one of three triggers: anomaly detection, recognition of a changed threat landscape, or the reporting of a monitored threat. Regardless of the origin, these paths lead to an employee creating a security ticket. This ticket initiates the execution and documentation of a risk analysis, which involves assessing the incident's scope and performing a risk assessment to evaluate the security risk. Completing the risk analysis sub-process (see 9 leads to the decision point, "Action Required?" If the answer is "No," the incident is immediately closed as resolved. If an escalation is necessary, the security firm is notified, resulting in a separate end event. If measures are required ("Yes"), the process branches into three independent, parallel paths: defining measures, implementing immediate corrective actions, and securing evidence. Implementing immediate corrective actions involves determining if the risk is normal or existential and executing a workaround or urgent system shutdown accordingly. The defined measures and immediate actions are then consolidated and fed into the Communicate activity, which plans and executes both internal and external

[6] As we will mention in the discussion, we anonymized and checked all context information before we sent it to proprietary systems.

[7] Compliance is mandatory and refers to following external, legal, or regulatory rules, while conformance is voluntary and refers to meeting internal or industry standards, specifications, or expectations.

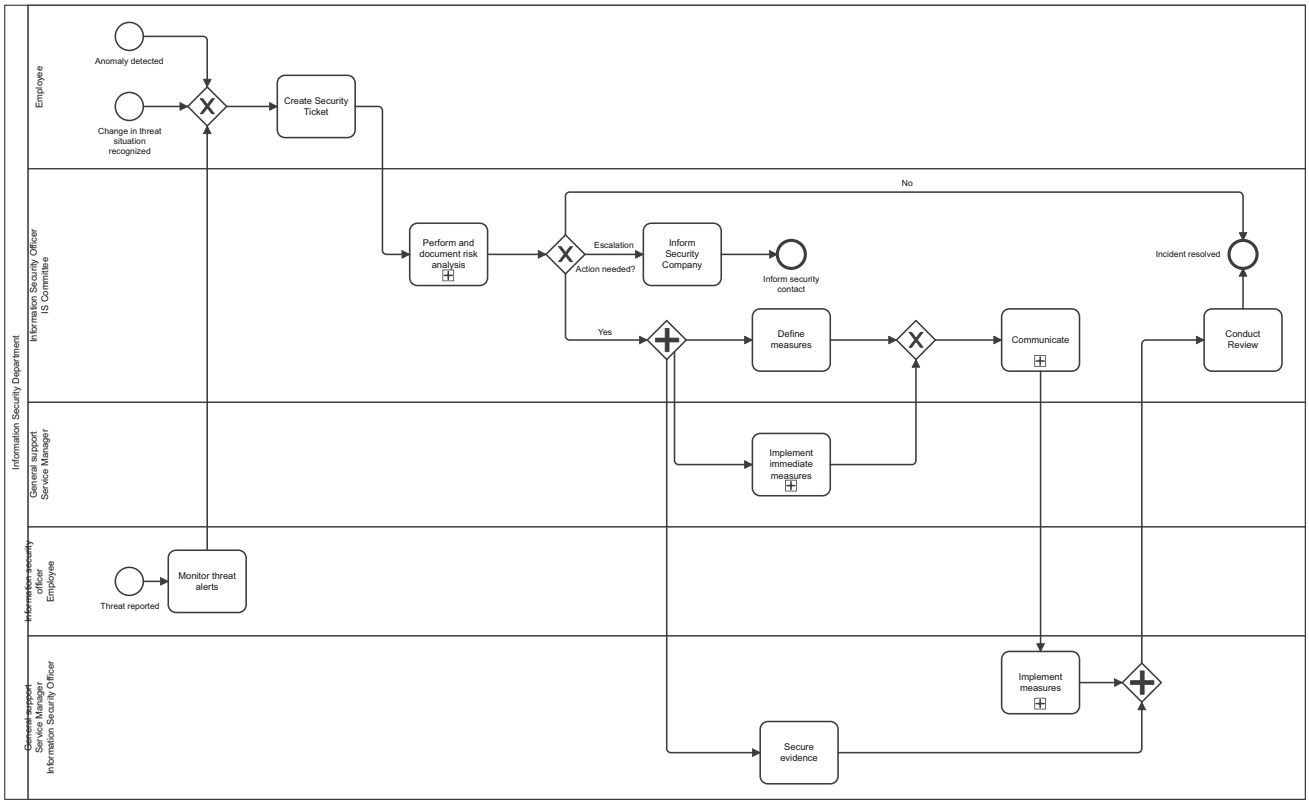

Fig. 3 Incident Response Process Model (expanded sub-processes in Section 9)

communication as needed, while evidence gathering continues in parallel. Following communication, the final measures are implemented, a step that determines the measure's scope and executes either customer-related protection or internal corrective actions. Finally, all parallel processes converge for the concluding activity, Conduct Review, before the entire process ends with the Incident Resolved status.

This process model is modeled on BPMN and does not include any ISO 27001-specific notation extensions. To verify whether the companies' incident response process fulfills the controls of ISO 27001, we derive one question for each of the 93 controls in the Annex of ISO 27001 within our framework. ISO 27001 generally outlines the requirements for information security management to ensure information security within an organization. Because of this, the factual (formal) compliance with the complete norm cannot fully be proven by a single process. However, since this is the incident response process we are examining, we can derive a significant amount of information from it. The questions for conformity assessment systematically refer to all controls of ISO 27001, focusing on one security control per question and pairing them as closely as possible with the respective control. For example, we transferred control 7.1. of the norm, which describes physical security perimeters, in the context of the question *"Are security perimeters defined and used to protect areas that contain information and other associated assets?"* or 5.26, which describes the response to information security incidents in the following question: *"Are information security incidents responded to in accordance with documented procedures?"*.

The next step was to apply the different contexts to the questionnaire and focus on providing additional instructions to the LLM in the form of prompting strategies, as presented in Section 5.1.4. Initially, we set out to try all possible combinations of prompting strategies and different state-of-the-art models. Yet, we focus on five relevant combinations seen in Table 2: *nothing* (no external context, no prompting strategy, the base line performance of the model), *context* (only providing necessary context, check if specific context itself helps), *contextFewShot* (adding few-shot examples to the prompt in addition to the context to streamline responses), *contextCoT* (additionally including chain of thought instructions) and *all* (that additionally contains role assignment and format instructions)[8].

To conduct the overall evaluation, we used different combinations of the prompting strategies mentioned above. For the assessment of the results obtained by using these prompts, we followed a two-fold approach: using

[8] Approved results and plots are contained in the repository (see Section 5.2).

Table 2 Overview of prompting strategy combinations

Strategy	Context	Few Shot	Chain of Thought	Format	Role
nothing	○	○	○	○	○
context	●	○	○	○	○
contextFewShot	●	●	○	○	○
contextCoT	●	○	●	○	○
all	●	●	●	●	●

`gemini-2.0-flash` (cf. [69]) to evaluate the performance of the model's answer, and providing the sample answer (*LLM-as-a-Judge*, cf. [20]), but also manually randomly checking the correctness based solely on question, model answer, sample answer and process model, without further knowledge of the prompting strategy for a small sample. After initial testing and trials, we stayed at the default parameters for the respective LLM models to not unnecessarily scale up the evaluation even more[9]. Here, it is essential that the generalizability and ongoing development and improvement of LLM models will benefit our approach, as we utilize them as tools that can be further configured. Compared to locally runnable models, these proprietary models have larger context lengths (here: one million) and better overall performance. Gemini was selected as one of the top-performing models, offering a competitive price-to-token ratio. After that, for *LLM-as-a-Judge* we introduced four different possible answers: *incorrect* if the given answer is incorrect in terms of content or does not relate to the question, *partially correct* if the answer provided contains generally correct explanations but draws incorrect conclusions, *mostly correct* if the answer provided only explains minor details incorrectly or if correct explanations are provided but a minimally incorrect conclusion is drawn, and *fully correct* if the answer provided gives a correct explanation and conclusion.

6.1.2 Findings

The results shown in Figure 4 demonstrate promising overall outcomes. With increased complexity and additional contextual information, LLMs generally perform better. While the first approach, without any additional information on the norm, performed quite poorly with just 19.4% fully correct responses, adding additional information step by step increases answer quality and accuracy to a maximum of 69.9%. As explained above, we allow the second LLM to actively differentiate not just on a binary basis, but also to include additional information in the responses that can be used by a human afterwards to guide decision-making, even when the result itself is ambiguous, unclear, or incomplete.

[9] For Gemini the temperature defaults to 1.0 ($\in [0; 2]$) which is comparable to $0.6 - 0.7$ ($\in [0; 1]$) with other models ($topP = 0.95$, $topK = 64$).

Current challenges with the use of LLMs are tied to the context length of the process models: context lengths and results are often limited due to computational resources and model specifications, as well as current payment models being fully based on token counts. For this, we also examined the different token lengths of the complete LLM communication and provided an overview in Figure 5. As seen here, the process model itself, which is contained in the 'nothing' strategy, takes up the largest part of the prompt. Adding the ISO 27001 increases the token count by approximately 12,500, while all other specifications for the prompt have a relatively negligible impact on the token count.

Looking directly at the results from the evaluation, the overall usability of general-purpose models as a tool using contextual information and process models provides a good fit for compliance checking and process model explanation. Using the "all" strategy, which includes all parts in the prompt, for the question *"Is information relating to information security threats collected and analyzed to produce threat intelligence?"*, the LLM correctly describes the sub-process "Security-Issue" and answers both parts of the question separately: first, the collection of information is described, and then the task of analyzing threats is mentioned. Since both requirements of the question are fulfilled, the LLM answers the question as "correct". Another question from the catalog, *"Are information security policies and topic-specific policies defined, approved, published, communicated, acknowledged, and reviewed at planned intervals and upon significant changes?"* was answered with "Yes". Justified by exemplary guidelines and policies for different domains of the added context information, the LLM correctly concludes that these documents are relevant policies. The question *"Is equipment sited securely and protected?"* was answered with "Yes" by the LLM. Although it cannot be answered, as no such information is provided based on the process model or context information. The only justification given is the text of ISO 27001 pertaining to Control 7.8.

6.2 Checking process compliance with the IEC 62443

The ISO 27001 evaluation demonstrated that our framework can successfully integrate unstructured regulatory documentation with process models to answer compliance-focused

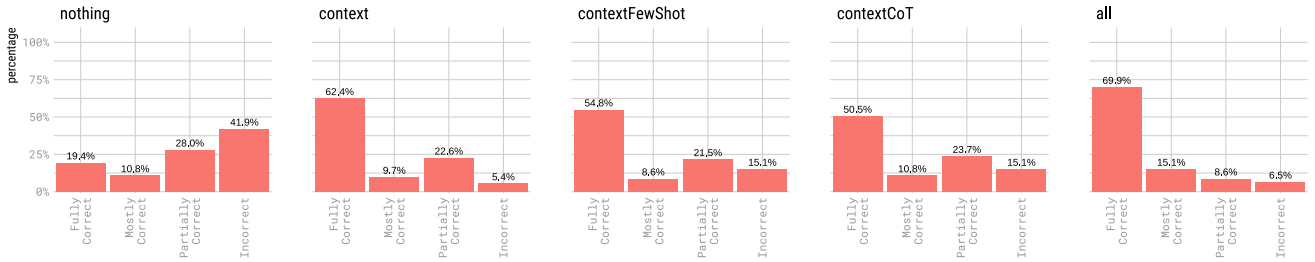

Fig. 4 Comparison of different prompting strategy success question catalog for the ISO27001

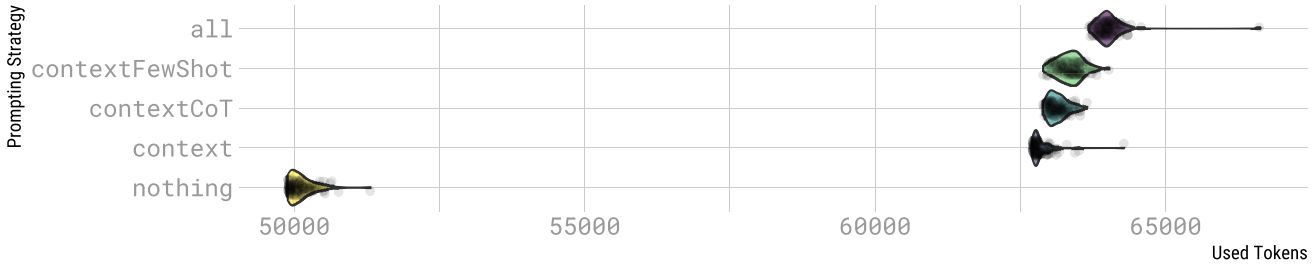

Fig. 5 Overview of token usage

queries, achieving up to 69.9% accuracy with the *all* prompting strategy. However, this use case primarily validated the framework's ability to retrieve and reason over external standards provided as separate contextual documents, while the process model itself contained minimal embedded security information. To complement this evaluation and test the versatility across different integration paradigms, we conducted another use case using IEC 62443, a standard specifically designed for Industrial Automation and Control Systems (IACS) security [36]. Unlike ISO 27001, which addresses broad organizational information security management applicable across diverse sectors, IEC 62443 provides prescriptive, OT-specific security requirements for industrial environments such as manufacturing and critical infrastructure. This focus on industrial systems presents unique, process-oriented challenges that have been the subject of ongoing research [10]. More critically for our evaluation purposes, the IEC 62443 use case employs a BPMN process model already enriched with structured security annotations through the SIREN (Security IoT Process Notation) BPMN extension [36], which represents security controls directly within the process model rather than as separate documentation. This architectural difference enables us to assess whether our framework can handle *structured-to-structured* integration (embedded BPMN security extensions mapped to standard requirements) as effectively as the *unstructured-to-structured* paradigm tested in the ISO 27001 case. Additionally, the IEC 62443 evaluation incorporates more rigorous dual validation, with a comprehensive manual expert assessment complementing the automated LLM-as-a-Judge methodology. This enables us to systematically compare both evaluation approaches in terms of reliability and identify their respective limitations in specialized compliance domains.

6.2.1 Structure

The specific requirements modeled are derived from IEC 62443-3-3, which defines system-level security requirements and security levels for industrial systems. It introduces seven Fundamental Requirements (FRs) that address core security aspects: *Identification and Authentication Control, Use Control, System Integrity, Data Confidentiality, Restricted Data Flow, Timely Response to Events*, and *Resource Availability*. Each FR is detailed through specific system requirements that can be implemented to achieve different Security Levels (SLs). These levels range from *SL 0* (no specific requirements) to *SL 4* (protection against attackers with sophisticated means and high motivation). This structured approach enables SIREN to use standard BPMN elements to explicitly represent security controls (such as indicating whether a sent message is encrypted) and to support the automated generation of monitorable rules for intrusion detection systems.

The modeled industrial process depicted in Figure 6 represents a real-world production workflow encompassing both manufacturing and quality control activities. The process's objective is to ensure that only defect-free components proceed to subsequent stages, using annotated IEC 62443-3-3 requirements to be met during implementation to ensure security. The process begins with the placement of a component onto a conveyor belt, followed by transportation to a preheating oven, where it is heated to 85°C and subse-

quently cooled. After this step, the component is transferred to the potting cell, where a specific mass is added, and a height inspection verifies the correctness of the filling. Components meeting the required quality criteria are placed onto a goods carrier and reheated. The component is then cooled for four hours before proceeding to the next stage. In contrast, components with excessive filling are classified as defective. These defective components trigger an automated message containing the unique component ID and are routed to a reject box, marking the end of the process. The process involves interaction between humans and various machines and integrates security requirements derived from the IEC 62443-3-3 standard. These requirements are explicitly modeled using SIREN as text annotations within the process model. For example, the functional requirement FR 1 – "Identification and Authentication Control" – is applied to the task "Heat component in preheating oven (85°C)", specifying that only authorized employees are permitted to execute this activity. To support this requirement, organization-specific attributes such as `is_human`, `user_id_exists`, or `id_is_unique` are attached to the task along with their respective values. These attributes are critical for fulfilling FR 1 of IEC 62443-3-3. For instance, `is_human: true` ensures verification that the logged-in user is a human operator, while `user_id_exists: true` confirms the presence of a valid user ID. Similar functional requirements, also shown in Figure 6, define security constraints and corresponding attributes for other tasks and machines involved in the process.

For the second use case, we went back to the drawing board as the BPMN extension itself adds additional overhead, and the specificity of the IEC 62443 would probably lead to different results compared to the first use case. For this, we began the evaluation of this use case specific SIREN extension as required by IEC 62443-3. As a domain-specific external context, add specific requirements of the standard to the prompt as implemented above. For an initial approach, we choose a naive prompting strategy by just providing the external context with zero-shot prompting to a locally running `deepseek-r1-0528-qwen3-8b`[10] model and asking two types of questions: One for model understanding and analysis that concern the general understanding and explanation of process models (*"What is the basic procedure of the process?", "Which participants (pools) are defined in the process and what roles do they take on?"*), and one specifically asking for the conformity of the modeled security standards with the standard (*"Which Security Level (SL) does the system achieve for Security Requirement (SR) 1.1 Identification and authentication of human users in accordance with IEC 62443-3-3?"*). We chose this model as it is publicly available for self-hosting and was one of the top-performing models with eight billion parameters at the time of running the evaluations in Q3 2025[11] The answers show promising results and affirm that the context is explicitly taken into account, providing the initial validation of our approach.

Based on these inputs and settings, we generated the following output, as presented in Figure 7. This demonstrates that the question's wording is considered and that context is explicitly taken into account when answering, providing the basis for validating our approach.

As seen above, the choice of a prompting strategy can significantly impact the quality of the results, as mentioned above. To achieve this, we created an additional questionnaire comprising 71 questions[12], along with sample answers and brief examples for each question. In addition to the questions above, we divided these into 20 questions that assess understanding and analysis of the process model itself as a starting point, and the remaining questions verify conformity with IEC 62443-3. We manually derived the sample answers from the relevant sections of the standard and the associated process model. The questions vary in wording and the aspects they address, covering various components of the process model and specific aspects of the context. Some of these questions focus on a general understanding of the process model, while others are specific to certain parts and focus on particular aspects of their content. The questions for conformity assessment systematically refer to all components of IEC 62443-3, focusing on a separate security requirement per question. Initial tests identified some points where the LLM-as-a-Judge approach might not work as expected with more focused context information. For this, we additionally manually evaluated the LLM answers and compared them against the fully automatic results. To quantitatively measure the performance of each prompting strategy and the level of agreement between the two evaluation methods, we employ several statistical techniques. We use classification metrics, such as precision and recall, to assess response accuracy and calculate Cohen's Kappa to evaluate inter-rater reliability between manual and automated evaluations. We kept the same LLM models to avoid additional parameters and kept the same parameter values.

6.2.2 Findings

Similar to the use case above, we also set out to use the questionnaire and the LLM-as-a-Judge approach, using the same prompting strategies and models, but with the model

[10] Using default parameters after initial testing: $temp = 0.6$ and $topP = 0.95$

[11] https://huggingface.co/deepseek-ai/DeepSeek-R1-0528-Qwen3-8B.

[12] The complete questionnaire and results are available in the repository, linked above.

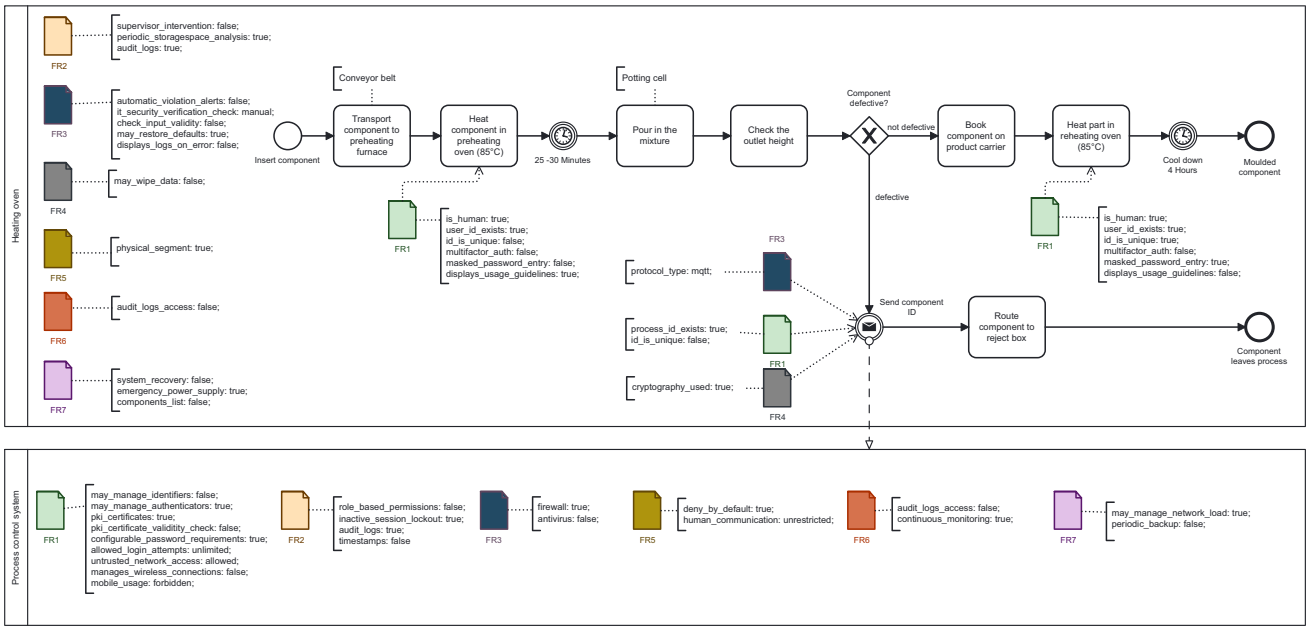

Fig. 6 The modeled IIoT-process extended with the SIREN notation [36]

```
"Here's a breakdown of the process flow, focusing on the key activities and sequence of events:
1.  **Start:** The process begins with the 'Insert component' start event.
2.  **Transport:** The component is transported to a preheating furnace via the 'Transport component to
↪  preheating furnace' task.
3.  **Preheating:** The component is heated in the preheating oven at 85°C ('Heat component in preheating
↪  oven (85°C)').
4. [...]"
```

```
"Based on the provided BPMN model and the associated text annotations, here's an analysis of the system's
↪  security level regarding SR 1.1 (Identification and authentication of human users) according to IEC
↪  62443-3-3.
**Relevant Text Annotations (Heating Oven): `is_human: true`, `user_id_exists: true`, `id_is_unique: false`,
↪  `multifactor_auth: false`, `masked_password_entry: false`, `displays_usage_guidelines: true`
**Analysis Based on IEC 62443-3-3 SR 1.1 Requirements:** Here's how the described system stacks up against
↪  typical requirements for different Security Levels (SL) regarding user identification and
↪  authentication. [...]"
```

Fig. 7 Exemplary answers of the LLM for model understanding and conformity verification

shown in Figure 6, which includes annotations concerning IEC 62443-3-3.

Figure 8 shows the results of different prompting strategies for the same model answers for manual and automatic evaluation. While the initial results are promising, with an average of 60% and 50% fully correct answers, the difference between manual and LLM-based evaluation is notable. Adding more and more contextual information, either directly to the prompt or by enhancing it using the strategies, does improve the overall results. Additionally, there is a significant difference between model understanding and analysis questions and verification of conformity. Whereas the former is based on general knowledge, the latter focuses specifically on the external context and is expected to perform worse initially. We will focus on the differences between manual and LLM-based evaluation with Figure 10.

To provide a rigorous quantitative analysis, we first present the statistical measures of our evaluation in Table 3.

The data illustrates the effectiveness of our framework. The baseline strategy, `nothing`, achieved accuracy of only 49.3%. In contrast, simply adding context (`context`) boosted accuracy to 70.4%, and the most effective strategy, `contextFewShot`, reached 74.6%. Furthermore, the metrics for the `Fully Correct` class are consistently strong for all context-aware strategies. The `context` strategy achieved precision of 0.95, indicating that its correct answers are highly reliable. The consistently high F1-Scores (≈0.89) confirm a robust balance between precision and recall.

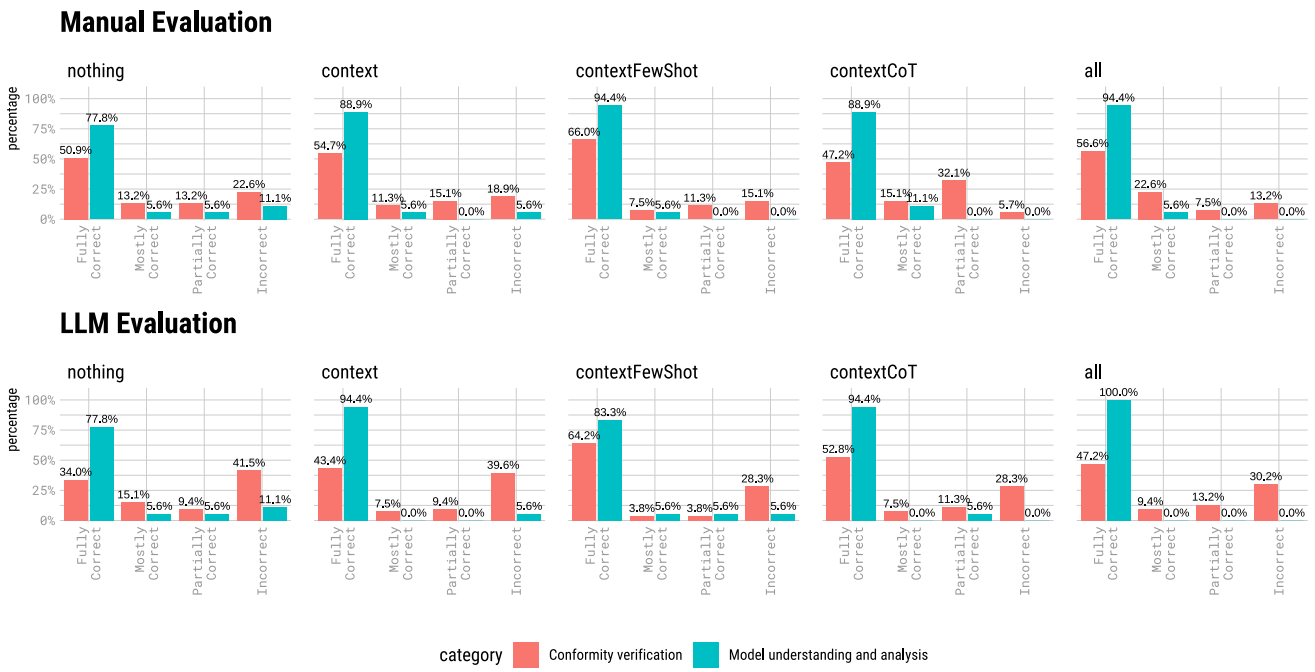

Fig. 8 Comparison of different prompting strategy success for manual and LLM evaluation of question catalog

Table 3 Detailed performance and agreement metrics for prompting strategies

Strategy	Acc.	Prec.	Rec.	F1	p
`nothing`	0.4930	0.8750	0.6829	0.7671	0.0805
`context`	0.7042	0.9500	0.8433	0.8941	0.1149
`contextFewShot`	0.7465	0.9184	0.8654	0.8911	0.2518
`contextCoT`	0.6344	0.8444	0.9268	0.8837	0.4561
`all`	0.6620	0.9302	0.8511	0.8889	0.0671

Note: Accuracy is based on manual evaluation.
Precision, Recall, and F1-Score refer to the 'Fully Correct' class

An additional goal was to validate the LLM-as-a-Judge method in the context of BPM, and our analysis reveals a clear picture. The overall T-test over all included prompting strategies indicates a statistically significant difference between manual and LLM evaluations ($p = 0.0015$), yet the overall weighted Cohen's Kappa of $\kappa = 0.3909$ and a phi coefficient [70] of $\varphi = 0.3973$ suggests a "fair" level of inter-rater agreement (cf. [71]). However, this is moderated by a crucial finding at the strategy-specific level. As shown in Table 3, the T-test for every prompting strategy yielded a p-value greater than 0.05. This suggests that, although there is an overall discrepancy, the difference between human and LLM evaluators is not statistically significant when a consistent prompting methodology is used. This supports the viability of using an LLM-as-a-Judge for scalable evaluation, provided the prompting is well-engineered.

For a more detailed analysis, we can examine individual question-level results for each evaluation type, as shown in Figure 9. Here we can see a heatmap that contains all evaluation results for each question, the evaluation types, and all selected prompting strategies. This figure reinforces the previous results and demonstrates the overall improvement and success of utilizing the additional provided context and enhanced prompting strategies for our approach. In general, here as well, the model understanding and analysis (encoded within the general "LLM-knowledge") are better than using external context, but the results seem promising. Upon closer examination of questions Q23 ("*What security level does the system achieve for SR 1.3 User account management in accordance with IEC 62443-3-3?*") and Q55 ("*What security level does the system achieve for SR 4.1 Confidentiality of information in accordance with IEC 62443-3-3?*"), where the model answer was deemed incorrect by the evaluating LLM but was mostly correct when manually evaluated, we observe the following: The first question considers the user account SL in accordance with the norm; the evaluation model correctly explains all requirements but then returns the wrong level, which is deemed incorrect by the LLM evaluation but

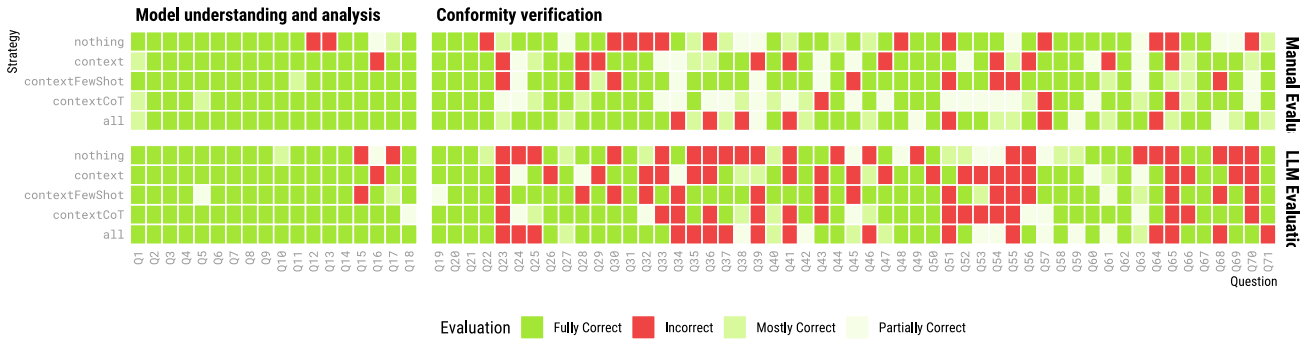

Fig. 9 Overview of results for both evaluation types based on questions directly

factually provides accurate information for a human expert. The other question focuses on the confidentiality of information when sending the component ID. Similarly, the model responses contain information not relevant to the step, which hinders correct evaluation but is factually correct.

Finally, for a direct comparison of both evaluation approaches, we can look at the differences shown in Figure 10: While many evaluation results (i.e., judging the correctness of the contextualized model answer based on the sample answer) are similar, especially for more complex questions, the results differ significantly. One example of this is Q24, where the knowledge and understanding of a human evaluation exceeds automated evaluation: For the question, *"What SL does the system achieve for SR 1.4 Identification Management according to IEC 62443-3-3?"*, a detailed explanation of the LLM correctly concludes that the model achieves SL1 based on the given variables, which is classified as "Fully Correct" in the manual evaluation. However, since the sample answer uses only a single variable to justify the achieved SL1, the answer is classified as "Incorrect" in an automated evaluation, even though it provides a correct explanation and conclusion in terms of content.

To provide a deeper understanding of the model's performance limitations beyond simple accuracy metrics, a detailed manual analysis was conducted on all responses not rated as "Fully Correct" or "Mostly Correct." This process involved categorizing the specific type of error the LLM made in each of the 86 instances classified as "Partially Correct" or "Incorrect," thereby identifying the root causes of failure. For this analysis, we established a four-part error typology:

- **Reasoning Failure**: The model successfully retrieves the correct pieces of information from both the process model and the standard, but fails to logically connect them to arrive at the correct conclusion.
- **Input Interpretation Failure**: The model misunderstands a part of the input, for instance, misinterpreting the function of a BPMN element or the meaning of a security requirement in the prompt.
- **Context Retrieval Failure**: The model fails to locate or use relevant information explicitly available within the provided context, such as a specific clause in the IEC 62443 standard or an attribute in the process model.
- **Hallucination**: The model invents information not present in the provided context or its general knowledge base.

The results of this analysis, visualized in Figure 11, reveal a clear and compelling distribution of these error types. Most notably, outright hallucination was the least frequent error, occurring in only two cases. In stark contrast, the most significant source of failure was Reasoning Failure, accounting for 49 instances, followed by Input Interpretation Failure (12) and Context Retrieval Failure (23).

This detailed breakdown offers a critical interpretation of the challenges inherent in using LLMs for process verification. The low incidence of hallucination suggests that in a context-rich environment, the model is less prone to inventing facts. Instead, the primary impediments are more fundamental. The widespread presence of Reasoning Failure indicates that the core difficulty is not a lack of information but a deficiency in the complex cognitive task of mapping the semantics of a structured BPMN model to the rules of an unstructured external standard.

Furthermore, the significant number of Input Interpretation and Context Retrieval Failures underscores the challenge of processing long, complex prompts. The model can either misunderstand the input's structure or fail to attend to the relevant information within the large context window, a problem consistent with the known "lost in the middle" limitation of transformer architectures. These findings suggest that future efforts to improve performance should shift the focus from the common narrative of preventing hallucinations to the more nuanced challenges of enhancing LLMs' logical synthesis, input parsing, and attention mechanisms for complex, domain-specific tasks.

Similar to the above, we again take a look at token counts for the different prompts. For this purpose, we first examined the distribution of token consumption for the different lan-

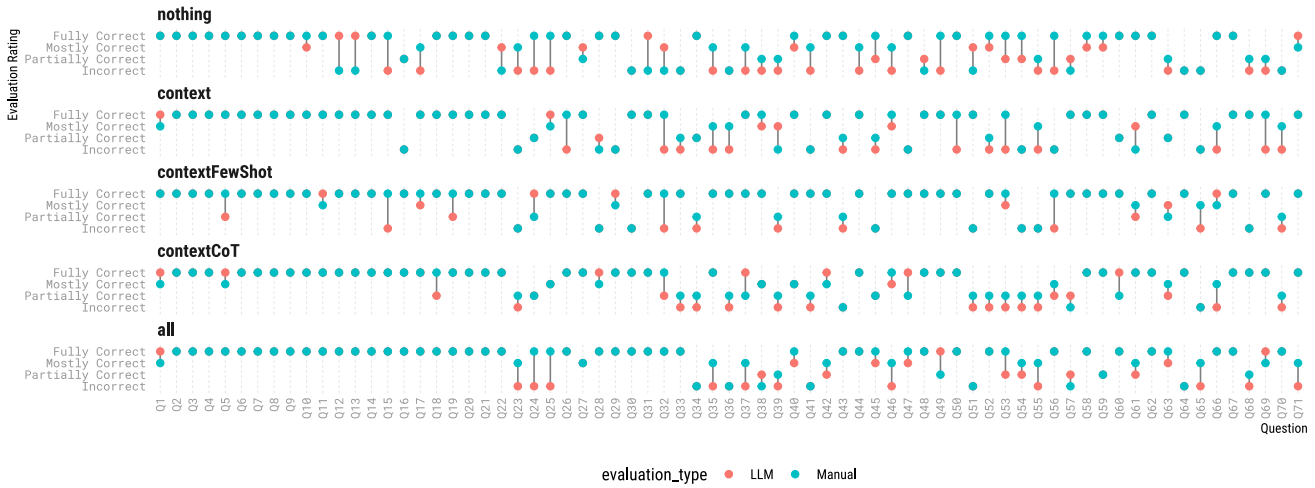

Fig. 10 Comparison of evaluation results from manual and automated evaluation of question catalog

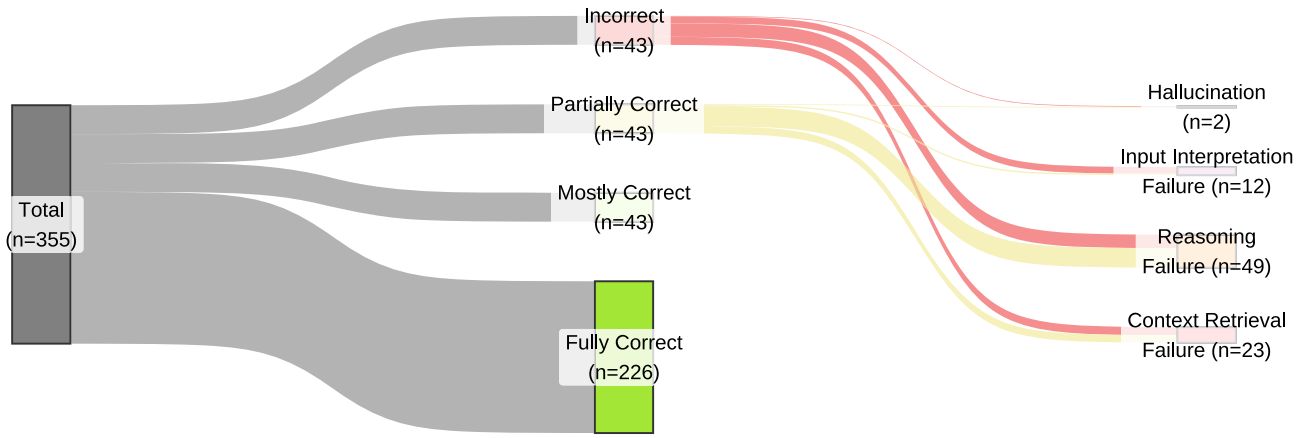

Fig. 11 Distribution of Incorrect Classifications by LLM

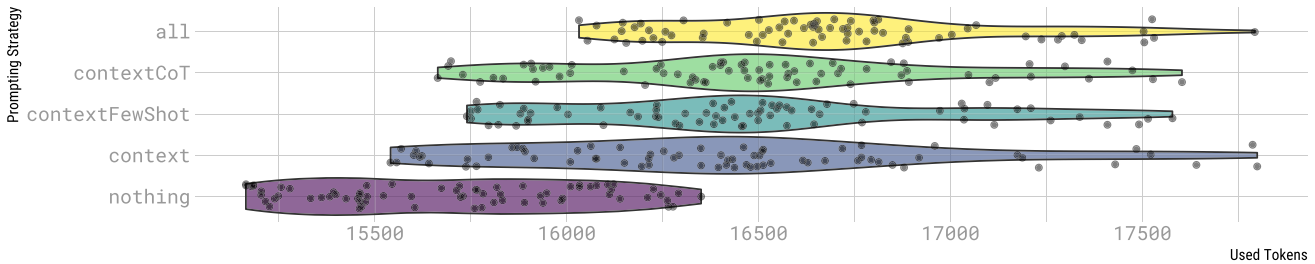

Fig. 12 Used tokens per prompting strategy for the evaluation

guage models in Figure 12. Thisillustrates the distribution of token consumption for the language model's responses across five distinct prompting strategies. Baseline consumption of approximately 15,000 tokens is visible across all conditions, corresponding to the fixed size of the input process model[13]. The variation in distributions primarily reflects the token cost of the generated answer, influenced by the prompting strategy.

The *nothing* strategy, representing the simplest prompt, establishes a baseline output distribution centered on 15,000 to 16,500 tokens. Progressively adding context (from *context* to *contextFewShot* and *contextCoT*) systematically shifts the distribution to the right, indicating an increase in the length of the generated output. The *all* strategy, which combines all contextual elements, yields the widest and most varied

[13] Manually checking the token lengths for the used process model results in around 15,000 tokens for the IEC62443 process model and 45,000 tokens for the ISO 27001 process model.

distribution, with token usage exceeding 17,500. Looking at the peaks in the *context* distribution, we can see that different lengths of the context are added to the prompt. A crucial finding is the apparent decoupling of output length from output quality (longer added context does not seem to influence result quality). The model's ability to generate a correct response appears to be independent of the verbosity induced by the prompt strategy.

7 Discussion

LLMs can effectively integrate external knowledge for process model compliance assessment, achieving accuracy up to 74.6% when properly contextualized with domain-specific standards. Through dual validation across ISO 27001 information security processes and IEC 62443-enriched Industrial IoT workflows, we demonstrate that LLMs can bridge the critical gap between structured process models and unstructured regulatory requirements. This capability represents a practical step toward operationalizing Security by Design principles in complex, regulation-heavy environments. Our evaluation across 71 questions revealed substantial variation in performance across prompting strategies. The *contextFew-Shot* configuration achieved the highest accuracy (74.6%), significantly outperforming baseline approaches and validating our hypothesis that systematic integration of external context with exemplar-based guidance enhances LLM reasoning for compliance tasks. This finding extends previous work on BPM-LLM integration [18, 45], demonstrating that while prompting strategies are important, the combination of structured process representations with unstructured domain knowledge is crucial for specialized compliance verification.

The statistical analysis revealed moderate but significant agreement between automated LLM-as-a-Judge evaluation and human expert assessment (Cohen's $\kappa = 0.3909$, $p < 0.001$), alongside strong classification performance (F1 $\approx$ 0.89). These metrics indicate that while automated evaluation shows promise for scalability, substantial discrepancies remain that warrant careful interpretation and human oversight—a finding with important implications for the broader application of LLM-based evaluation frameworks in BPM. Importantly, our framework successfully handled real-world process complexity: the IEC 62443 use case involved the BPMN model with security-specific extensions (SIREN notation [36]), while the ISO 27001 case required reasoning across 93 Annex A controls. The ability to maintain contextual coherence across such varied regulatory landscapes demonstrates the framework's generalizability beyond single-standard applications.

A central contribution of this work is our systematic error typology, derived from analyzing 78 incorrect responses from the IEC 62443 use case. We identified four distinct failure modes: *Reasoning Failure* ($n = 49$), *Context Retrieval Failure* ($n = 23$), *Input Interpretation Failure* ($n = 4$), and *Hallucination* ($n = 2$). This distribution reveals a critical insight: contrary to prevalent concerns about LLM hallucination [23], the dominant challenges in compliance verification are *logical synthesis* and *attention mechanisms* for retrieving relevant contextual segments. Reasoning Failures occurred when the model retrieved correct information but failed to apply appropriate logical operations or draw valid inferences required for compliance assessment. This pattern suggests that current LLMs, while excellent at pattern recognition and retrieval, struggle with the multi-step deductive reasoning chains characteristic of regulatory compliance verification. Context Retrieval Failures, meanwhile, manifested when models either overlooked relevant passages in the provided standards or fixated on irrelevant sections–highlighting limitations in attention mechanisms when processing lengthy, structured documents, such as ISO 27001 (spanning 93 controls) or IEC 62443-3-3.

The relative rarity of direct hallucinations ($n = 2$) in our study contrasts with the general behavior of LLMs on open-domain tasks, likely attributable to our Context Integration Prompting strategy, which explicitly grounds model responses in the provided documentation. This finding reframes the challenge from one of preventing fabrication to one of enhancing structured reasoning. These error patterns align with observations from related BPM-LLM work. Similar to [44], who reported precision and recall limitations in document classification despite targeted prompting, our findings suggest that document-to-reasoning tasks require architectural advances beyond prompt engineering alone. However, our F1 score of 0.89 substantially exceeds reported performance on comparable classification tasks, likely because our structured BPMN input reduces ambiguity compared to free-text organizational documents.

Our dual-evaluation approach revealed a critical limitation of automated assessment frameworks that has important implications for the broader adoption of LLM-as-a-Judge methodologies in specialized domains. While Cohen's $\kappa = 0.3909$ indicates fair agreement between LLM-as-a-Judge and human experts, qualitative analysis uncovered a fundamental flaw: the evaluator model exhibited rigidity in recognizing valid alternate reasoning paths. The most illustrative case occurred in Q24, where the LLM-as-a-Judge incorrectly classified a detailed, manually verified correct response as "Incorrect" because the reasoning path differed from the sample answer, even though it reached the same valid conclusion through sound logic. This failure mode, in which automated evaluation penalizes valid methodological diversity, has serious implications for unsupervised compliance verification. In regulatory contexts, where multiple valid interpretations of standards coexist (particularly across IEC 62443's four SLs), such inflexibility renders fully auto-

mated evaluation unsuitable without human validation of the evaluation framework itself. The results from both use cases confirm our hypothesis on the general applicability of enriching process models with external context in combination with LLMs and support the reliability of the presented framework and the approach. The classification performance is strong, although limitations arising from LLM-as-a-Judge became apparent in isolated cases during manual review, as discussed above. Nevertheless, human review remains useful and but also underlines the general reliability of the generated responses.

This finding contributes to ongoing debates in the LLM evaluation literature [19, 20] by demonstrating that evaluation model performance cannot be assumed from benchmark tasks–domain-specific validation remains essential. For the BPM community, this suggests that while LLM-as-a-Judge approaches offer scalability advantages over manual assessment, they require careful calibration and periodic human auditing, particularly for nuanced compliance tasks where reasoning diversity is inherent. Although this dual-validation requirement increases deployment complexity, it enhances reliability, making it a necessary trade-off for high-stakes cybersecurity applications.

A critical challenge emerged during implementation: achieving necessary performance required large-context proprietary APIs (e.g., Google Gemini), necessitating transmission of sensitive process models to third-party cloud services. For enterprises operating under strict data confidentiality requirements, this external dependency often proves infeasible under existing security policies (particularly in critical infrastructure sectors governed by IEC 62443). This constraint motivated our exploration of local model alternatives, though performance degradation was observed with smaller open-source models.

Our work advances the state of knowledge established in our systematic literature review (Table 1) in several key dimensions. Unlike document classification approaches as shown in [44], which reported suboptimal performance with generic prompting, our framework achieved F1 $\approx$ 0.89 by systematically integrating structured process models with external contextual standards. This performance difference underscores the value of *structured-unstructured data fusion* over purely text-based approaches. Compared to [45], which propose a prompting method for process redesign from event logs, which noted significant output quality variation based on prompt specificity, our modular prompting architecture (combining role assignment, chain-of-thought, and context integration) provides more stable performance across diverse question types. Where the authors of [18] and [15] explored RAG-based approaches for BPM decision support, our framework extends this paradigm specifically to compliance verification. Our error typology further reveals that while RAG successfully addresses retrieval (Context Retrieval Failures represent only 29% of errors), reasoning architecture remains the primary bottleneck. Crucially, our work addresses the research gap identified in Section 4.3: existing methods for security compliance checking (formal verification, model checking, Security-by-Contract frameworks) rely solely on structured data, whereas LLM-based approaches primarily operate on unstructured text. Furthermore, by demonstrating a viable LLM-based framework for integrating standards like IEC 62443-3, our work provides a novel, automated approach to addressing the "perspectives and challenges" of process-oriented IIoT security management, as articulated in [10, 11].

While our findings demonstrate the viability of LLM-based compliance verification, several limitations warrant consideration. The computational requirements of our approach pose practical barriers. Token consumption of 50,000–65,000 per query for comprehensive standards such as ISO 27001 requires state-of-the-art models with extended context windows, thereby excluding smaller or older architectures. Additionally, the verbosity of XML-formatted BPMN models contributes to token consumption and may hinder input interpretation, as evidenced by Input Interpretation Failures in our error analysis. Our dual-evaluation design, although rigorous, primarily concentrated manual validation on the IEC 62443 use case due to resource constraints. Although the ISO 27001 case relied more heavily on LLM-as-a-Judge evaluation, our identification of systematic biases in automated assessment (Q24 alternate reasoning path issue) suggests manual validation across both cases would strengthen generalizability claims. It should be noted that uncertainty due to incomplete or overly abstract process models and information, as well as the vagueness and ambiguity of annotations, can deeply impact the results. In addition to subjective modeling decisions, changing requirements and human error can also negatively affect the quality of the results, but are not directly connected to the approach at hand.

Returning to our research question (*How can Large Language Models be leveraged to integrate external knowledge for the explanation and assessment of process model compliance against security standards?*) our findings demonstrate this is achievable with accuracy up to 74.6% through systematic integration of structured BPMN models, unstructured regulatory documentation, and carefully designed prompting strategies. However, success requires awareness of specific failure modes (particularly limitations in Reasoning and Context Retrieval), of human validation infrastructure (especially for calibrating the evaluation framework), and of computational resource considerations (such as token costs and context window requirements). The novel error typology we introduce provides a diagnostic framework for improving LLM-based compliance systems and

reveals that reasoning architecture, rather than hallucination prevention, constitutes the primary technical challenge.

While our results demonstrate the practical viability of augmented compliance workflows, they also underscore that human experts remain crucial, not only for validating model outputs but also for validating evaluation methodologies, interpreting edge cases, and providing organizational context that transcends documented standards. This finding aligns with broader themes in AI-augmented professional work: automation excels at scaling routine analysis, but human judgment remains essential for nuanced interpretation and accountability. Furthermore such compliance checking assessments enable early-stage risk prevention and provide the necessary framework to establish robust preventive security measures which is central for cybersecurity.

Ultimately, this work provides a methodological foundation and practical demonstration that context-aware LLM integration can support more transparent and proactive Security by Design practices, thereby bridging the critical gap between operational process models and the complex regulatory landscapes they must navigate.

8 Conclusion and future work

In this study, we have designed and validated a generic framework for enriching business process models with external, unstructured context to enhance their explanation and verification. To evaluate the practical applicability of this framework, we instantiated and tested it with a specific use case from the domain of Industrial Internet of Things (IIoT) security, using the IEC 62443-3 standard as the external contextual information. Our results demonstrate that this approach is effective, showing that integrating external context, especially when combined with multiple prompting strategies, significantly enhances the quality and accuracy of LLM-generated responses.

Building on the success of this framework, future research could focus on creating more powerful human-AI collaborative systems for process management by addressing two interconnected dimensions: error mitigation and confidence quantification. The error typology established in this work provides a foundation for developing targeted correction strategies. For reasoning errors, multi-step prompting techniques could be implemented that require LLMs to explicitly cite standard references before formulating compliance judgments, thereby enforcing traceable logical chains. To address input interpretation errors arising from complex BPMN paths, preprocessing techniques that decompose intricate process structures into manageable segments merit exploration for more reliable analysis (see above).

Beyond error correction, an equally critical avenue is the development of methods to quantify model output confidence, enabling systems to automatically flag uncertain or ambiguous assessments. This confidence-aware approach would enable process analysts to strategically focus their expert validation efforts on the most complex or questionable compliance checks, rather than reviewing all outputs uniformly.

The long-term vision for this research is a system that operates with even greater autonomy. Future work will focus on developing a framework where the LLM can actively query a corporate knowledge base to find and retrieve relevant external context, such as standards, regulations, or policies, for any given process model, rather than relying on manually provided information. This autonomous capability, potentially incorporating dynamic or real-time data, would revolutionize how organizations manage compliance and understand their process landscape. As developments in LLM research continue (model size, context windows, and multi-modal capabilities, or thinking models that further generally improve LLM results), the potential of this approach grows. Seeing LLMs as a new tool for NLP tasks within BPM offers a fresh perspective on structuring complex environments, helping to automate and support currently manual tasks that are critical to successful value creation.

9 Subprocesses of Figure 6

See Figs. 13, 14, 15 and 16.

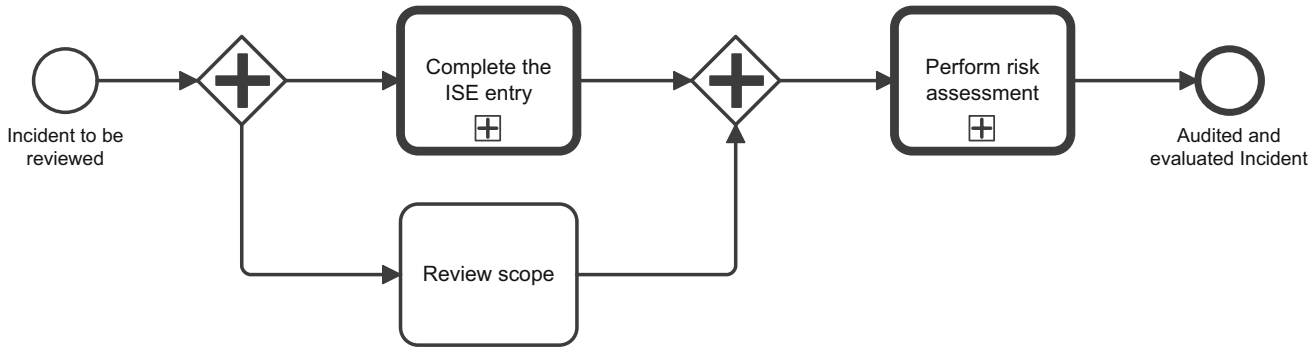

Fig. 13 Perform and document risk analysis Subprocess

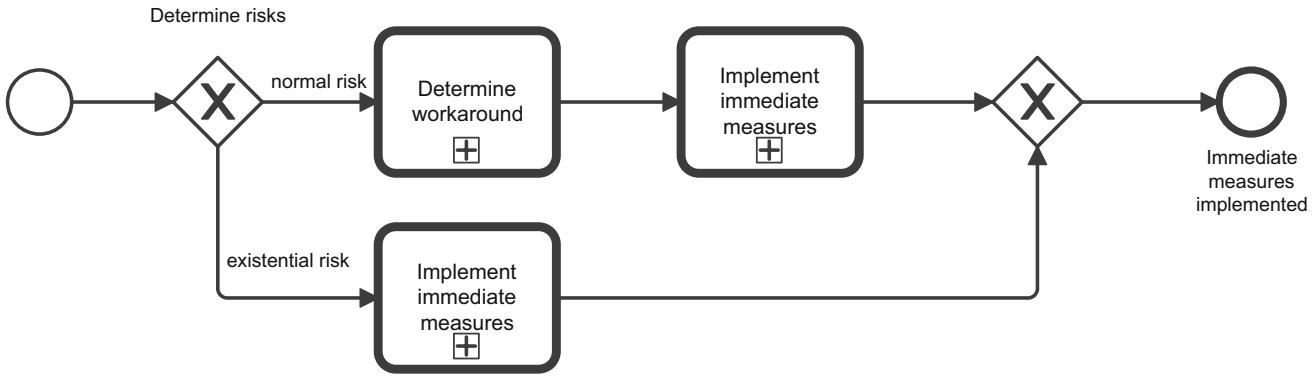

Fig. 14 Implement immediate measures Subprocess

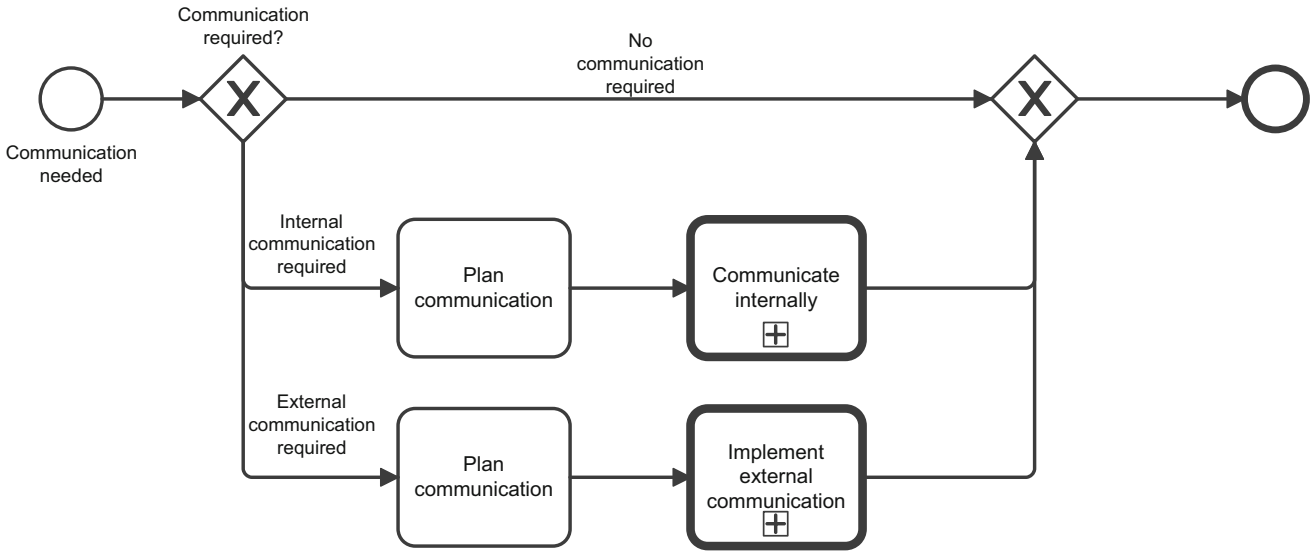

Fig. 15 Communicate Subprocess

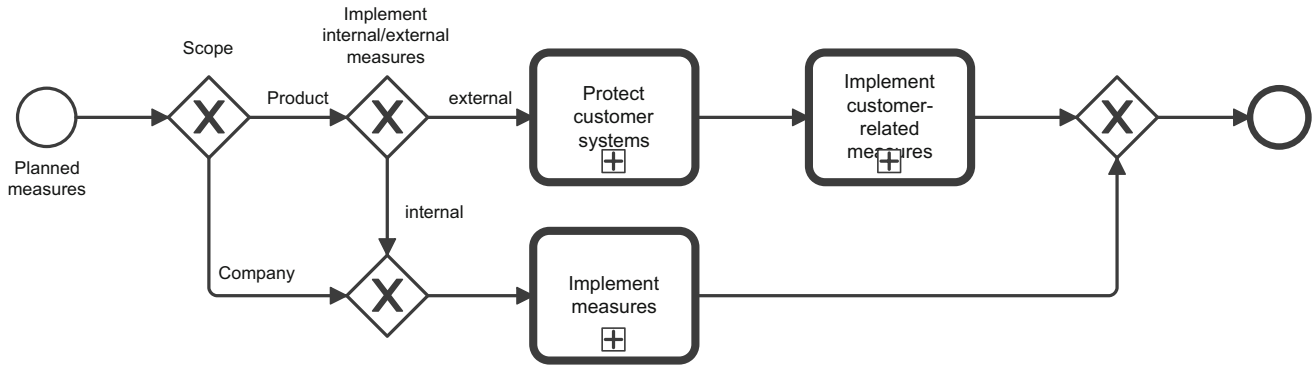

Fig. 16 Implement measures Subprocess

Author Contributions L.K. and L.P. wrote the main manuscript text in consultation with S.S.. All authors reviewed the manuscript.

Funding Open Access funding enabled and organized by Projekt DEAL. This work is funded by the "Bavarian Ministry of Economic Affairs, Regional Development and Energy" within the project Security Iiot pRocEss Notation (SIREN) (Grant No. DIK-2308-0012//DIK0511).

Data Availability No datasets were generated or analysed during the current study.

Declarations

Competing interests The authors declare no competing interests.

References

1. International Organization for Standardization and International Electrotechnical Commission. ISO/IEC 27001:2022 – Information security management systems – Requirements. ISO/IEC, Geneva, Switzerland, 2022. Fourth edition. Available from www.iso.org
2. International Electrotechnical Commission. IEC 62443-3-3:2013 – Industrial communication networks – Network and system security – Part 3-3: System security requirements and security levels. IEC, Geneva, Switzerland, 2013. First edition. Available from www.iec.ch
3. Masys, A.J.: Security by Design: Innovative Perspectives on Complex Problems. Springer International Publishing (2018)
4. Casola, V., De Benedictis, A., Rak, M., Villano, U.: A novel security-by-design methodology: Modeling and assessing security by slas with a quantitative approach. J. Syst. Softw. **163**, 110537 (2020)
5. Kianpour, M., Raza, S.: More than malware: unmasking the hidden risk of cybersecurity regulations. International Cybersecurity Law Review **5**(1), 169–212 (2024)
6. Beāte Krauze. An analysis of resilience in digital business ecosystems. In Research Challenges in Information Science. RCIS 2025., page 162–171. Springer Nature Switzerland, 2025
7. Stenmark, D.: Leveraging tacit organizational knowledge. J. Manag. Inf. Syst. **17**(3), 9–24 (2000)
8. vom Brocke, J., Baier, M.-S., Schmiedel, T., Stelzl, K., Röglinger, M., Wehking, C.: Context-aware business process management. Bus. Inform. Syst. Eng **63**(5), 533–550 (2021)
9. Abowd, G.D., Dey, A.K. Brown, P.J.: Nigel Davies, Mark Smith, and Pete Steggles. Towards a better understanding of context and context-awareness. In: Handheld and Ubiquitous Computing, pages 304–307. Springer Berlin Heidelberg, (1999)
10. Stefan Schönig, Markus Hornsteiner, and Christoph Stoiber. Towards process-oriented iiot security management: Perspectives and challenges. In BPMDS/EMMSAD@CAiSE, volume 450 of Lecture Notes in Business Information Processing, pages 18–26. Springer, (2022)
11. Markus Hornsteiner, Linda Kölbel, Daniel Oberhofer, and Stefan Schönig. A reflection on process-oriented industrial iot security management. In ICISSP (1), pages 242–253. SCITEPRESS, 2025
12. Han van der Aa, Josep Carmona Vargas, Henrik Leopold, Jan Mendling, and Lluís Padró. Challenges and opportunities of applying natural language processing in business process management. In COLING 2018: The 27th International Conference on Computational Linguistics: Proceedings of the Conference: August 20-26, 2018 Santa Fe, New Mexico, USA, pages 2791–2801. Association for Computational Linguistics, 2018
13. Noah Ziems and Shaoen Wu. Security vulnerability detection using deep learning natural language processing. In IEEE INFOCOM 2021 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), page 1–6. IEEE, May 2021
14. Jie Zhang, Haoyu Bu, Hui Wen, Yongji Liu, Haiqiang Fei, Rongrong Xi, Lun Li, Yun Yang, Hongsong Zhu, and Dan Meng. When llms meet cybersecurity: a systematic literature review. Cybersecurity, 8(1), February 2025
15. M. L. Bernardi, A. Casciani, M. Cimitile, and A. Marrella. Conversing with business process-aware large language models: the BPLLM framework. Journal of Intelligent Information Systems, 2024
16. P. Kogler, W. Chen, A. Falkner, A. Haselböck, and S. Wallner. Modelling Engineering Processes in Natural Language: A Case Study. In 28th ACM International Systems and Software Product Line

Conference, pages 170–178, Dommeldange, Luxembourg, 2024. ACM
17. Michael Grohs, Luka Abb, Nourhan Elsayed, and Jana-Rebecca Rehse. Large Language Models Can Accomplish Business Process Management Tasks, page 453–465. Springer Nature Switzerland, 2024
18. T. Kampik, C. Warmuth, A. Rebmann, R. Agam, L. N. P. Egger, A. Gerber, and M. Weidlich. Large Process Models: A Vision for Business Process Management in the Age of Generative AI. KI - Künstliche Intelligenz, pages 1–15, 2024
19. Li, D., Jiang, B., Huang, L., Beigi, A., Zhao, C., Tan, Z., Bhattacharjee, A., Jiang, Y., Chen, C., Tianhao, W., Kai Shu, L., Cheng: and Huan Liu. Opportunities and challenges of llm-as-a-judge, From generation to judgment (2025)
20. Jiawei, G., Jiang, X., Shi, Z., Tan, H., Zhai, X., Chengjin, X., Li, W., Shen, Y., Ma, S., Liu, H., Wang, S., Zhang, K.: Yuanzhuo Wang. Lionel Ni, and Jian Guo. A survey on llm-as-a-judge, Wen Gao (2024)
21. A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, and I. Polosukhin. Attention is all you need. Advances in Neural Information Processing Systems, 30, 2017
22. W. X. Zhao, K. Zhou, J. Li, T. Tang, X. Wang, and Y. Hou. A survey of large language models. arXiv preprint arXiv:2303.18223, 1(2), 2023
23. M. U. Hadi, R. Qureshi, A. Shah, M. Irfan, A. Zafar, M. B. Shaikh, et al. A survey on large language models: Applications, challenges, limitations, and practical usage. Authorea Preprints, 3, 2023
24. Fan, L., Li, L., Ma, Z., Lee, S., Yu, H., Hemphill, L.: A bibliometric review of large language models research from 2017 to 2023. ACM Transactions on Intelligent Systems and Technology **15**(5), 1–25 (2024)
25. Weske, M.: Business Process Management, 3rd edn. Springer, Berlin Heidelberg (2019)
26. M. Dumas, L. M. Rosa, J. Mendling, and A. H. Reijers. Fundamentals of Business Process Management. Springer, 2018
27. Bernardo, R., Galina, S.V.R., Dallavalle, S.I., de Pádua: The BPM lifecycle: How to incorporate a view external to the organization through dynamic capability. Bus. Process. Manag. J. **23**(1), 155–175 (2017)
28. Marek Szelągowski. Evolution of the bpm lifecycle. In federated conference on computer science and information systems, pages 205–211, 2018
29. Wil, M.P.: van der Aalst. Data Science in Action. Springer, Process Mining (2018)
30. Ivan Compagnucci, Flavio Corradini, Fabrizio Fornari, Andrea Polini, Barbara Re, and Francesco Tiezzi. A systematic literature review on IoT-aware business process modeling views, requirements and notations. Softw. Syst. Model., 2022
31. Poss, L., Schönig, S.: Location-aware business process modeling and execution. Softw. Syst. Model. **24**(1), 37–67 (2024)
32. Koh Song Sang and Bo Zhou. Bpmn security extensions for healthcare process. In 2015 IEEE International Conference on Computer and Information Technology; Ubiquitous Computing and Communications; Dependable, Autonomic and Secure Computing; Pervasive Intelligence and Computing. IEEE, 2015
33. Jan vom Brocke, Alan Hevner, and Alexander Maedche. Introduction to Design Science Research, pages 1–13. Springer International Publishing, Cham, 2020
34. Hevner, A.R., March, S.T., Park, J., Ram, S.: Design Science in Information Systems Research. MIS Q. **28**(1), 75 (2004)
35. Peffers, K., Tuunanen, T., Rothenberger, M.A., Chatterjee, S.: A design science research methodology for information systems research. J. Manag. Inf. Syst. **24**(3), 45–77 (2007)
36. Markus Hornsteiner and Stefan Schönig. Siren: Designing business processes for comprehensive industrial iot security management. In Aurona Gerber and Richard Baskerville, editors, Design Science Research for a New Society: Society 5.0, pages 379–393, Cham, 2023. Springer Nature Switzerland
37. Julian Neuberger, Lars Ackermann, Han van der Aa, and Stefan Jablonski. A universal prompting strategy for extracting process model information from natural language text using large language models. In Conceptual Modeling, pages 38–55. Springer Nature Switzerland, 2025
38. Konstantinos Sintoris and Kostas Vergidis. Extracting business process models using natural language processing (nlp) techniques. In 2017 IEEE 19th Conference on Business Informatics (CBI), volume 01, pages 135–139, 2017
39. Ana Cláudia de Almeida Bordignon, Lucinéia Heloisa Thom, Thanner Soares Silva, Vinicius Stein Dani, Marcelo Fantinato, and Renato Cesar Borges Ferreira. Natural language processing in business process identification and modeling: A systematic literature review. In Proceedings of the XIV Brazilian Symposium on Information Systems. Association for Computing Machinery, 2018
40. C. Okoli. A guide to conducting a standalone systematic literature review. Communications of the Association for Information Systems, 37, 2015
41. Pedro Antonio Boareto, Anderson Luis Szejka, Eduardo Freitas Rocha Loures, Fernando Deschamps, and Eduardo Alves Portela Santos. Accelerating industry 4.0 and 5.0: The potential of generative artificial intelligence. In International Conference on Innovative Intelligent Industrial Production and Logistics, pages 456–472. Springer, 2024
42. Alina Hafner, Holger Wittges, and Stefanie Rinderle-Ma. Genai in business process management: A systematic review of the current state. In AMCIS 2025 Proceedings, number 9. Association for Information Systems (AIS), 2025
43. M. J. Page, J. E. McKenzie, P. M. Bossuyt, I. Boutron, T. C. Hoffmann, C. D. Mulrow, and D. Moher. The prisma 2020 statement: an updated guideline for reporting systematic reviews. BMJ, n71, 2021
44. Ilagan, J.R., Ilagan, J.B., Basallo, C.L., Alabastro, Z.M.: Exploratory prompting of large language models to act as co-pilots for augmenting business process work in document classification. Procedia Computer Science **237**, 420–425 (2024)
45. K. Lashkevich, F. Milani, M. Avramenko, and M. Dumas. LLM-Assisted Optimization of Waiting Time in Business Processes: A Prompting Method. In Business Process Management, volume 14940 of Lecture Notes in Computer Science, pages 474–492. Springer Nature Switzerland, 2024
46. R. Paulose and V. Neelanath. Generative ai-driven automation of business process reimagination. In 2024 IEEE Recent Advances in Intelligent Computational Systems (RAICS), pages 1–6, Kothamangalam, Kerala, India, 2024. IEEE, IEEE
47. Yue Wang, Ningyuan Yi, Wenjing Chang, and Jianjun Yu. Leveraging llms for automated compliance risk identification in business processes. In Mufti Mahmud, Maryam Doborjeh, Kevin Wong, Andrew Chi Sing Leung, Zohreh Doborjeh, and M. Tanveer, editors, Neural Information Processing, pages 456–471, Singapore, 2025. Springer Nature Singapore
48. José Luis Cobo-Ariza, Joaquín Arregui, Antonia M. Reina Quintero, Ángel Jesús Varela-Vaca, and María Teresa Gómez-López. Explaining the compliance of security policies for gdpr in business processes. In Jānis Grabis and Yves Wautelet, editors, Advanced Information Systems Engineering Workshops, pages 355–367, Cham, 2025. Springer Nature Switzerland
49. Jinying Li and Ananda Maiti. Applying large language model analysis and backend web services in regulatory technologies for continuous compliance checks. Future Internet, 17(3), 2025
50. A. Beheshti, J. Yang, Q. Z. Sheng, B. Benatallah, F. Casati, S. Dustdar, and S. Xue. ProcessGPT: Transforming Business Process Management with Generative Artificial Intelligence. In 2023 IEEE

International Conference on Web Services (ICWS), pages 731–739, Chicago, IL, USA, 2023. IEEE, IEEE
51. B. Estrada-Torres, A. del Río-Ortega, and M. Resinas. Mapping the Landscape: Exploring Large Language Model Applications in Business Process Management. In Enterprise, Business-Process and Information Systems Modeling, volume 511 of Lecture Notes in Business Information Processing, pages 22–31. Springer Nature Switzerland, 2024
52. Humam Kourani, Alessandro Berti, Daniel Schuster, and Wil M. P. van der Aalst. Process Modeling with Large Language Models, page 229–244. Springer Nature Switzerland, 2024
53. Licardo, J.T., Tanković, N., Etinger, D.: A method for extracting bpmn models from textual descriptions using natural language processing. Procedia Computer Science **239**, 483–490 (2024)
54. M. Minor and E. Kaucher. Retrieval augmented generation with llms for explaining business process models. In Case-Based Reasoning Research and Development, volume 14775 of Lecture Notes in Computer Science, pages 175–190. Springer Nature Switzerland, 2024
55. V. Pasquadibisceglie, A. Appice, and D. Malerba. Lupin: A llm approach for activity suffix prediction in business process event logs. In 2024 6th International Conference on Process Mining (ICPM), pages 1–8, Kgs. Lyngby, Denmark, 2024. IEEE, IEEE
56. J. Schnepf, T. Engin, S. Anderer, and B. Scheuermann. Studies on the use of large language models for the automation of business processes in enterprise resource planning systems. In Natural Language Processing and Information Systems, volume 14762 of Lecture Notes in Computer Science, pages 16–31. Springer Nature Switzerland, 2024
57. S. Softic, E. Lüftenegger, D. Resanovic, S. Softic, and A. Popan. Leveraging sentiment analysis and reporting for re-designing business processes using large language models: A sentipromo case study in airline check-in processes. In Advances in Production Management Systems. Production Management Systems for Volatile, Uncertain, Complex, and Ambiguous Environments, volume 731 of IFIP Advances in Information and Communication Technology, pages 3–17. Springer Nature Switzerland, 2024
58. M. Toslali, E. Snible, J. Chen, A. Cha, S. Singh, M. Kalantar, and S. Parthasarathy. Agrabot: Accelerating third-party security risk management in enterprise setting through generative ai. In Companion Proceedings of the 32nd ACM International Conference on the Foundations of Software Engineering, pages 74–79, Porto de Galinhas Brazil, 2024. ACM
59. Maxim Vidgof, Stefan Bachhofner, and Jan Mendling. Large language models for business process management: Opportunities and challenges. In Business Process Management Forum, volume 490 of Lecture Notes in Business Information Processing, pages 107–123. Springer Nature Switzerland, 2023
60. Ziche, C., Apruzzese, G.: Llm4pm: A case study on using large language models for process modeling in enterprise organizations. In: International Conference on Business Process Management. volume 527 of Lecture Notes in Business Information Processing, pp. 472–483. Springer, Springer Nature Switzerland (2024)
61. Barrientos, M., Winter, K., Rinderle-Ma, S.: Impact analysis of regulatory requirement changes on business process compliance. Inf. Softw. Technol. **194**, 108079 (2026)
62. Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M.: Wen tau Yih. Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive nlp tasks, Tim Rocktäschel (2021)
63. Sander Schulhoff, Michael Ilie, Nishant Balepur, Konstantine Kahadze, Amanda Liu, Chenglei Si, Yinheng Li, Aayush Gupta, HyoJung Han, Sevien Schulhoff, Pranav Sandeep Dulepet, Saurav Vidyadhara, Dayeon Ki, Sweta Agrawal, Chau Pham, Gerson Kroiz, Feileen Li, Hudson Tao, Ashay Srivastava, Hevander Da Costa, Saloni Gupta, Megan L. Rogers, Inna Goncearenco, Giuseppe Sarli, Igor Galynker, Denis Peskoff, Marine Carpuat, Jules White, Shyamal Anadkat, Alexander Hoyle, and Philip Resnik. The prompt report: A systematic survey of prompt engineering techniques, 2024
64. Venable, J., Pries-Heje, J., Baskerville, R.: Feds: a framework for evaluation in design science research. Eur. J. Inf. Syst. **25**(1), 77–89 (2016)
65. Sonnenberg, C., vom Brocke, J.: Evaluations in the Science of the Artificial – Reconsidering the Build-Evaluate Pattern in Design Science Research, page 381–397. Springer, Berlin Heidelberg (2012)
66. Hubert Österle, Jörg Becker, Ulrich Frank, Thomas Hess, Dimitris Karagiannis, Helmut Krcmar, Peter Loos, Peter Mertens, Andreas Oberweis, and Elmar J Sinz. Memorandum on design-oriented information systems research. European Journal of Information Systems, 20(1):7–10, 2011
67. Roel, J.: Wieringa. Design Science Methodology for Information Systems and Software Engineering. Springer, Berlin Heidelberg (2014)
68. Ángel Jesús Varela-Vaca, María Teresa Gómez-López, Yolanda Morales Zamora, and Rafael M. Gasca. Business process models and simulation to enable gdpr compliance. International Journal of Information Security, 24(1), December 2024
69. Colin White, Samuel Dooley, Manley Roberts, Arka Pal, Ben Feuer, Siddhartha Jain, Ravid Shwartz-Ziv, Neel Jain, Khalid Saifullah, Siddartha Naidu, et al. Livebench: A challenging, contamination-free llm benchmark. arXiv preprint arXiv:2406.19314, 2024
70. Davide Chicco, Matthijs J Warrens, and Giuseppe Jurman. The matthews correlation coefficient (MCC) is more informative than cohen's kappa and brier score in binary classification assessment. IEEE Access, 9:78368–78381, 2021
71. J Richard Landis and Gary G Koch. The measurement of observer agreement for categorical data. biometrics, pages 159–174, 1977