Aaro Korpela

Framework for Secure Industrial Process Automation Using Local AI Agents: A Case Study on Hitachi's RTU Configuration Process

School of Technology and Innovations
Bachelor's Thesis
Industrial Engineering and Management

Vaasa 2025

UNIVERSITY OF VAASA
School of Technology and Innovations
Author: Aaro Korpela
Title of the thesis: Framework for Secure Industrial Process Automation Using Local AI Agents: A Case Study on Hitachi's RTU Configuration Process: A Case Study on Hitachi's RTU Configuration Process
Degree: Bachelor of Science in Tech
Degree Programme: Industrial Engineering and Management
Supervisor: Binod Timilsina
Year: 2025 **Pages:** 59

ABSTRACT:
Industrial process automation has developed immensely, and the current Artificial Intelligence (AI) boom brings in new possibilities for companies. However, integrating cloud-based AI solutions is challenging for companies operating with critical infrastructure, due to data security and privacy requirements of customer systems. Therefore, this thesis explores the feasibility of local and secure AI agent automation for industrial workflows, while specifically focusing on a case study of Remote Terminal Unit (RTU) configuration process at Hitachi Energy. The main research objective is to develop a framework for connecting industrial process analysis to local AI automation, thus, evaluating if an on-premises agent performs in RTU configuration tasks securely and effectively compared to traditional manual engineering.

Using a Design Science research methodology, this study employs a mixed-methods approach with Process Mining and Value Stream Mapping to identify automation candidates, which are then developed into a local AI agent artifact. This artifact is based on the Qwen2.5-coder:1.5B Small Language Model (SLM), which is fine-tuned and tested against manual workflows. Three subprocesses are tested: file management, credential management and backup generation. Quantitative findings show statistically significant performance improvement in the backup subprocess due to configuring parallelization, while file management shows reliable but comparable speed against the manual method. However, the agent fails in completion of the credential management subprocess, for limitations in its autonomous terminal operation.

Results of the thesis imply that local edge AI is viable and secure for industrial configuring applications. Additionally, the findings prove SLMs to be a more sustainable step towards Industry 5.0, by only consuming 0.037 Wh per query, than cloud-based LLMs, while keeping sensitive data safe locally. Despite these benefits, limitations regarding the agent's ability to handle multi-step workflows are acknowledged. Hence, future research directions are suggested to improve the interactive capabilities within the agent's tools, and to add additional agents with different roles and responsibilities to uncover the improvements which agentic AI could pose for industrial processes.

KEYWORDS: Industrial Automation, AI Agents, Cybersecurity, Edge Computing, Process Analysis, Small Language Models, Fine-tuning.

Contents

1 Introduction 7

1.1 Background of the study 7

1.2 Objectives and research questions of the study 8

1.3 Scope and structure of the research 9

2 Literature review 11

2.1 Process analysis and task identification 12

2.1.1 Business process modelling and notation 12

2.1.2 Process mining 14

2.1.3 Value stream mapping 16

2.2 Automation feasibility and levels of automation 19

2.3 AI fundamentals: local vs cloud-based AI agents 24

2.3.1 Key technologies for local industrial agents 25

2.3.2 Secure on-premises vs cloud-based deployment 28

3 Research methodology 32

3.1 Case description 33

3.2 Data collection methods 33

3.2.1 Process capture and analysis 34

3.2.2 Dataset and artifact creation 34

3.2.3 Testing procedure 35

3.3 Data analysis and metrics 35

3.4 Reliability and ethics 36

4 Results and findings 37

4.1 Process Mining and Value Stream Analysis 38

4.2 Quantitative findings 40

4.3 Qualitative findings 43

4.4 Fine-tuning results 43

5 Discussion and Conclusion 45

5.1 Discussion 45

5.1.1 Interpretation of results and relation to research questions 45
5.1.2 Comparison with existing literature and practical implications 48
5.1.3 Limitations and potential follow-up studies 49
5.2 Conclusion 52
5.2.1 Summary of research and answers to research questions 52
5.2.2 Significance, implications, and future research directions 52
References 54
Appendices 59
Appendix 1. File management performance snippet 59
Appendix 2. Power consumption during AI agent usage 59

Images

Image 1. Fine-tuned model’s terminal output on backup task. 44

Image 2. Base model’s terminal output on backup task. 44

Figures

Figure 1. BPMN 2.0 visualization of RTU firmware configuration process. 12

Figure 2. RTU engineering process modelled by VSM. 17

Figure 3. Fundamental architecture of an AI agent. 25

Figure 4. Cloud and edge AI's computational place. 29

Figure 5. Subprocess execution performance in mean time with standard deviation. 37

Figure 6. Process Mining map of webpage visits during configuration tests. 38

Figure 7. Process Mining heat map of time spent on webpages. 39

Figure 8. Value Stream Map of the test configuration process. 39

Figure 9. Fine-tuned the model's training loss over training steps. 43

Tables

Table 1. 10 Automation levels of decisions and actions (Parasuraman et al., 2000). 20

Table 2. Quantitative and qualitative variables were used in the study. 36

Table 3. File upload and update performance. 41

Table 4. User credential management performance. 41

Table 5. Backup generation performance. 42

Table 6. GPU power consumption per user query. 42

Abbreviations

AI = Artificial Intelligence

BPMN = Business Process Modelling and Notation

HMI = Human-Machine Interface

LAN = Local Area Network

LLM = Large Language Model

RAG = Retrieval-Augmented Generation

RPA = Robotic Process Automation

RTU = Remote Terminal Unit

SCADA = Supervisory Control and Data Acquisition

SLM = Small Language Model

VSM = Value Stream Mapping

1 Introduction

This thesis explores how industrial processes in companies can be automated using local Artificial Intelligence (AI) agents, with a case study of Hitachi Energy Finland Oy Grid Automation unit's remote terminal unit (RTU) configuration process. It is conducted as a case study. The research aims to review current methods and frameworks to analyze organizational processes to determine their flows and bottlenecks. Then, concepts behind local AI agents will be covered before conducting the case study to observe the feasibility of the current state of local AI agents for industrial process automation. The outcome of this study will benefit the understanding of process analysis and safe AI automation within the company.

1.1 Background of the study

The hunger for greater efficiency in the current industrial market has led to significant development of AI automation in rapid time. This change is happening in all company processes from procurement to shipping and so forth (Samuels, 2025). Thus, many repetitive and time-consuming processes in companies can be handled through AI agents. For organizations, this shift means that their employees have more time to focus on less repetitive tasks, reducing human errors (De Silva et al., 2025, pp. 2-4).

Therefore, company processes can be developed with the right analytical tools to find bottlenecks, and with the knowledge of what criteria or quality the process must have to be an AI agent automation prospect. For companies, current cloud-based AI automation solutions are in use, however their application to industrial processes is complex due to the sensitive data of customers, that is not allowed to leave the company's premises. Therefore, this thesis addresses the gap of AI implementation for processes concerning data security by proposing an alternative concept to industrial process automation through local AI agents.

Local solutions within enterprise premises are becoming more crucial when data handling and storage have stricter legislation, e.g., with the EU's GDPR (European Parliament & Council of the European Union, 2016). Thus, the development of secure local AI agent solutions that are run on-premises allow companies to never send their customer or internal data outside of the enterprise.

The RTU configuration process of Hitachi is an industrial process that is done in Supervisory Control and Data Acquisition (SCADA) projects. The process is done at the factory by engineers, and it is manual repetitive work, which is time-consuming and done one RTU at a time. These configuration tasks, as currently being done one-by-one, lower the possible capacity of configurable RTUs; thus, having a scalable AI agent managing the configuration tasks would increase the possible capacity and offer insight into how many RTUs an engineer is able to configure. Hence, developing the proper methods to analyze a process to find automatable tasks and the knowledge of how to implement local AI agents into industrial processes proves important for making them secure, scalable, and reducing the engineers' time consumption and human errors.

1.2 Objectives and research questions of the study

Thus, the overall objective of this thesis is to develop a framework that connects established industrial process analysis methods with the current capabilities of local AI agents. This provides a structured approach for identifying and implementing secure automation opportunities in industrial flows. The aim is to develop a local AI agent for RTU configuration for Hitachi's configuration stage as a case study to further evaluate and reflect on the industrial process AI agent automation framework. The thesis's research questions are listed below:

1. What tasks and process characteristics make industrial workflows feasible for local AI agent automation?

2. In which ways are local AI agents more secure and preserve privacy compared to cloud-based solutions?
3. How does the Hitachi case study's AI agent perform in RTU configuration compared to the traditional process of a human engineer?

The research questions are chosen, as they follow the logical flow of what knowledge is needed to improve industrial processes with local AI agent automation. Suitable industrial processes for AI agent automation can be found with the findings of the first research question. Then, addressing the critical security aspects of AI agents can be found from research question two, and answering the last research question validates the concept through a practical case study. Thus, the research questions are chosen for a logical flow into researching a way to implement AI agent automation into industrial processes.

1.3 Scope and structure of the research

The scope of the thesis is centered around developing the framework for finding and connecting the industrial processes that can be automated through local AI agents. It will dive into how the processes can be identified and what characteristics the process needs. The security factors of local AI agents will be discussed, and the fundamentals of AI agents will be covered. The scope of the thesis does not include the practical methodology of how the case study's AI agent is built, but it includes testing. Therefore, this research brings significance to industrial process analysis to develop the analysis tools for processes that can be improved with AI agent automation.

This thesis has five main chapters, where the first one introduces the topic by clarifying the significance of industrial process automation with AI agents, improving productivity, saving time, and reducing human errors. The second chapter continues with a literature review of existing studies. It also builds a theoretical framework for the concepts that are evaluated further in the case study. The third chapter reveals the research approach

and data collection methods. The fourth chapter will go over the findings of the case study, and the final fifth chapter will discuss, evaluate, and draw conclusions on the findings of the thesis.

2 Literature review

To establish the theoretical foundation for developing the automation framework, this section reviews the existing studies for analyzing industrial processes to identify tasks that are automation candidates with AI agents. Following Alotaibi (2020), business processes are defined as the set of coordinated activities performed by people or machines to achieve organizational goals. Related to Hitachi's remote terminal unit configuration workflow, a process refers to the sequential steps an engineer takes to configure RTUs. Tasks are activities that are the most minor identifiable units of work that contribute to the process, e.g., configuration steps (Dumas et al., 2020, p. 4; van der Aalst, 2022, p. 28). Then, those tasks that are identified as repetitive and rule-based, that are low in cognitive complexity, are considered as automation candidates (Choi et al., 2021, p. 10). Hence, the literature review's three frameworks will focus on the business processes tasks that demonstrate potential to be local AI agent automation candidates.

Business process analysis and task identification into automation candidates will be the first framework. Hence, the first framework connects to the first research question by examining what can be automated. After determining what can be automated, the second framework of automation feasibility and levels of automation reviews the methods for deciding how many automation candidates can be automated and to what extent. Finally, the third framework of AI fundamentals, with an in-depth dive into local versus cloud-based AI agents, examines how automation by local AI agents can be implemented securely for businesses.

These frameworks were chosen as they are well studied; however, there remains a research gap in integrating them together into a flow for secure industrial automation. The structure of the literature review follows chronologically.

2.1 Process analysis and task identification

To understand how industrial processes can be modelled and analyzed to identify tasks suitable for automation, this section reviews three complementary analytical methods: Business Process Modelling and Notation (BPMN), Process Mining and Value Stream Mapping (VSM). BPMN provides a standardized structure for visualizing processes for a clear and concise representation of workflows across systems and their stakeholders (Dumas et al., 2020, pp. 75-78). Process Mining develops this by discovering and validating the actual processes and their flows based on event log data, which helps in finding deviations from the process and further inefficiencies (van der Aalst, 2022, pp. 3-6). VSM brings in a dimension of lean manufacturing with an interest in the process flow for efficiency and waste (Ferreira et al., 2023, p. 82). These business process analysis tools are reviewed for their latest developments to act as the foundation for this thesis's process automation analysis framework.

2.1.1 Business process modelling and notation

Business Process Modelling and Notation is a standardized method of visualizing business processes and has been standardized as the IEC 19510:2013 (International Organization for Standardization, 2013). Therefore, BPMN is an internationally recognized standard in process modelling. Figure 1 presents an example of the RTU firmware configuration process modelled using BPMN 2.0 notation.

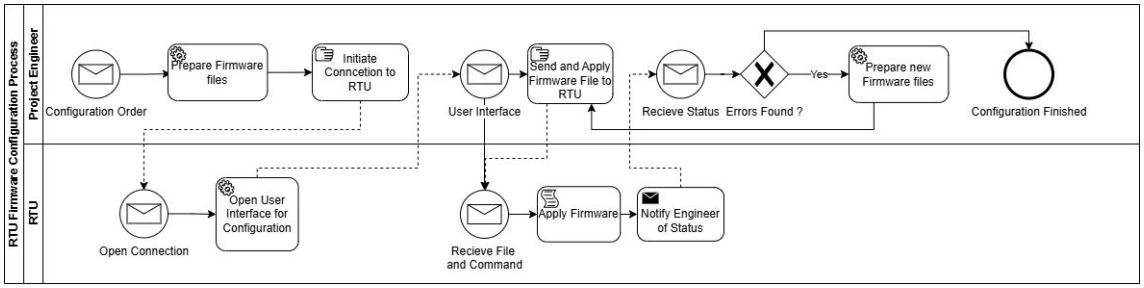

Figure 1. BPMN 2.0 visualization of RTU firmware configuration process.

While BPMN is used globally for industrial discrete process modelling, Strutzenberger, Mangler, and Rinderle-Ma (2024) have observed that many industrial processes are continuous in nature, e.g., flows in chemical engineering or control loops. Strutzenberger et al. (2024) propose BPMN extensions for continuous processes, such as loop control, timed measurement, and regulation. They also define execution semantics for BPMN extensions to ensure proper logical execution for the continuous processes that have been modelled, ensuring the checking and execution of extended BPMN models with an executable language, e.g., code or process-engine representation (Strutzenberger, Mangler, & Rinderle-Ma, 2024). Through use cases and interviews with industrial experts, the extensions of BPMN for continuous process modelling proved to be a key factor in BPMN's development (Strutzenberger et al., 2024). Although Strutzenberger et al. (2024) validated their BPMN extensions through interviews with industry experts, their evaluation was rudimentary due to their small interviewee sample, as they only had three groups of experts, where each group contained four experts. Extending BPMN modelling into industrial continuous processes facilitates the possibility of finding more automation candidates from repetitive processes, thus enhancing the tool's coverage for industrial process analysis.

In addition to extending BPMN to continuous processes, it has been developed for logical validation of BPMN processes, which has been studied by Corradini et al. (2021), who identified the problem that BPMN is a functional method of displaying processes that both technical and business personnel can understand. However, they discovered that faulty logic and that some processes do not reach end events, thus having incorrect logic in the BPMN process diagrams. They studied this problem by developing a Business Process Verification (BProVe) framework, which verifies BPMN collaboration models, e.g., models that have messaging between two IT systems (Corradini et al., 2021). With their framework, they were able to translate the graphical collaboration models into a formal representation by a MAUDE system, which expressed the models in process logic and a set of rules, further allowing the automatic inspection of it through model checking. After successfully checking the BMNP logic, its key properties, such as soundness,

safeness, and message consistency, can be ensured to be precise (Corradini et al., 2021). BProVe framework integrates MultiVeStA into its logical model checking, which estimates the property satisfaction through multiple simulated executions of the BPMN collaboration model. This dual method develops BPMN accuracy by allowing logical and simulated analysis of complex industrial processes, while Corradini et al. (2021) prove the theoretical validation for BPMN processes; their study does not assess the models' accuracy or performance in real industrial settings, hence only portraying theoretical proof for validating BPMN processes. Therefore, the BProVe framework bridges the gap between BPMN's visual modelling and formal process validation, supporting the use of BPMN for analyzing interorganizational industrial processes.

2.1.2 Process mining

Process Mining validates organizational processes and their flow based on event log data, by creating visual flow diagrams showing how organizations process flow in terms of time and efficiency (van der Aalst, 2022, pp. 3-6). Within the scope of Robotic Process Automation (RPA), Process Mining plays an important role in validating repetitive business processes with e.g., AI-supported systems (Chinthamu et al., 2025).

AI-based RPAs' optimization by Process Mining has been examined by Chinthamu et al. (2025). They had found a literature gap between AI-system RPA research and process mining techniques, as previous studies had been using RPA for business processes, which were recognized by themselves, thus hindering the impact of workflow automation that RPA has (Chinthamu et al., 2025). Therefore, they propose that applying Process Mining before RPA, as organizations can target the most efficient and optimizable workflows, as the study found a 25% increase in efficiency within the business processes in healthcare and customer service fields (Chinthamu et al., 2025). Even though the paper describes and does empirical experiments on RPA by using Process Mining, their sample sizes are never stated in the research, as they only state their findings in terms of percentages, thus hindering their credibility, as their sample size remains unknown. The paper promises to deliver a framework for an AI-system RPA, which is enhanced with a Process

Mining driven model, but it fails to deliver, as only a vague overall flowchart is provided in Figure 1 of the paper, thus developing an underdeveloped model. Nonetheless, the paper by Chinthamu et al. (2025) shows how Process Mining is a key factor in effective RPA for business processes with automation by AI-based systems, while its findings' reliability is hindered by the unknown sample size, the paper proves that Process Mining will act as an important tool for businesses to target their workflows for RPA.

Adding to the research on Process Mining's efficiency gains, Vulpe et al. (2022) bring evidence of using Process Mining in RPA solutions provided to industrial companies. Their research was based on Process Mining through UiPath's platform. The platform's consumer companies were leveraging their event log data into automation opportunities. UiPath's integration of Process Mining into its RPA solution platform provided the customer companies with the tools to gain end-to-end visibility and reduce operational costs by analyzing key performance indicators and return on investment statistics in their process flows (Vulpe et al., 2022). Hence, Process Mining in automatable tasks benefits from measurable metrics and develops the integration of Robotic Process Automation. Though as Vulpe et al.'s (2022) paper relies primarily on one industrial player, UiPath, with data and descriptive analysis, their findings open themselves to being biased, as the data comes from a company offering the service themselves. Thus, the paper's scientific thoroughness is weak due to the lack of independent validation, as the paper focuses more on the implementation rather than on the comprehensive evaluation of Process Mining being used across the industry's RPA providers. Nevertheless, the paper builds on the understanding of Process Mining as an analytical tool for automation by highlighting the operational and business value gains that companies acquire from RPA.

Supporting Vulpe et al. (2022) and Chinthamu et al. (2025), El-Gharib and Amyot (2023) conducted a systematic literature review on integrating RPA in industrial settings with Process Mining and the challenges faced in integration. The review was done with 32 papers published since 2018 on Process Mining and RPA, being highly relevant and offering breadth within the field. The paper found that Process Mining is applied

increasingly in process discovery stages for RPA, by either task mining, where software is run on employees' computers, which builds process maps, or process discovery algorithms from historical event log data (El-Gharib & Amyot, 2023). The review also concluded common difficulties in Process Mining for RPA, as El-Gharib and Amyot (2023) found that gathering data reliably from users and preprocessing historical event logs were difficult. Data gathering proved to be complex due to some industries using legacy systems, which had trouble integrating with the Process Mining software, and preprocessing noisy and long event logs required significant time and effort from the Process Mining data analysts. The paper shows a focused scope around its research questions, with challenges and integration stages of RPA with Process Mining. While the paper proves significant breadth in research between Process Mining and RPA, it lacks depth in the individual tools being used and the techniques, thus giving more of an overview of how Process Mining is being applied to RPA in the industry. Together with Chinthamu et al. (2025) and Vulpe et al. (2022), this study supports how the integration of Process Mining to RPA is well-established while being constrained by technical and data quality limitations.

2.1.3 Value stream mapping

One tool that Lean operations analysts continue to use is Value Stream Mapping (VSM), which was initially developed by Rother and Shook (1999). It is used to create flow diagrams of business processes with a focus on time that is actively spent on labor and what is spent waiting, thus allowing for steps in the process flow to be identified for possible improvements, e.g., being an automation candidate (Rosenberg, 2021, pp. 18-20). Figure 2 provides an overview of a typical VSM structure, showing how materials, information, and time flow using standard variables. VSM's recent developments have been in integration into Industry 4.0, sustainability, and social aspects in VSM (Batwara et al., 2023).

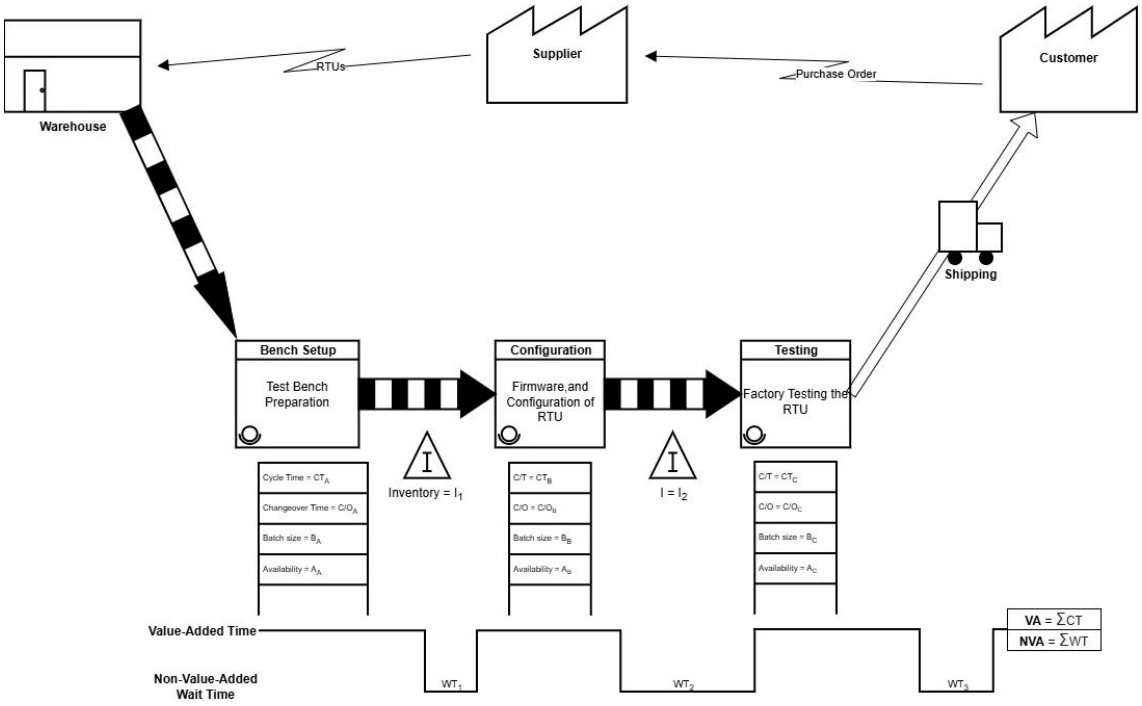

Figure 2. RTU engineering process modelled by VSM.

The literature review done by Batwara et al. (2023) on VSM discovered a rapid increase of research in VSM from 2008 to 2023 with developments in Industry 4.0, sustainability, and social aspects. The paper's research on the development of VSM's integration to Industry 4.0 discusses a research gap between integrating VSM process models into Internet of Things devices, as companies have an increasing trend in using (Batwara et al., 2023). The development of VSM in finding automation candidates from organizational processes uncovered a social aspect, as tasks that were previously performed by the human workforce were being automated and replaced by robots or software, thus uncovering a problem of how companies can effectively repurpose their workforce into new tasks to keep lean development socially acceptable within the companies (Batwara et al., 2023). Another dimension of VSM that has been getting more attention in research is sustainability and how it is incorporated as a measurable indicator in VSM and considered (Batwara et al., 2023). The review shows strength in covering the SCOPUS database and articles in English regarding the research of VSM; however, it lacks in source diversity, as it is centered around the Western databases, articles, and findings, having a small representation of how VSM is developing in the

global south. This limitation only shows how VSM frameworks are used and what its effect is on the global north and developed industries. In contrast, the global south and its less mature industries remain underexplored. Overall, the review proves how this classical Lean method is developing into a modern process-analysis method aligned with the new industrial trends of automation, Industry 4.0, and sustainability.

Building on Batwara et al.'s (2023) review, Reslan et al. (2025) introduce a framework that adapts the classical VSM model to Industry 4.0 through digital twin technology. In their paper, they conducted a foundational case study of constructing a simple two-part assembly line and created a digital twin VSM model to examine it. Through the digital twin, Reslan et al. (2025) were able to connect the real-time data automatically to the VSM of the assembly line and analyze it for waste through Lean manufacturing. In the case study, they were able to reduce their process lead time by 38 minutes (7%), thus demonstrating that as industries and their machines become more digitalized through Industry 4.0, there is significant value for process development in modelling the processes through digital twins, which are connected to real-time data sources (Reslan et al., 2025). However, the paper functions only as proof of concept by initiating the development of the framework for digital twin and VSM modelling of industrial processes, as the case study was a limited setup of an assembly line with only two robots and two other machines, hence requiring further studies in diverse and realistic industrial settings. Another challenge involved integrating data from robots and machines into the VSM model, as no standardized format existed, which required extensive data preprocessing. Altogether, the paper illustrates the future value of integrating Industry 4.0 into process analysis tools like VSM by enabling real-time data to be processed through digital twin models.

Al-Rifai's (2024) case study on applying VSM in an electronics manufacturer offers a practical solution to the social challenges identified by Batwara et al. (2023) in process streamlining. In the paper, VSM and Lean methods were used to drastically reduce production lead time by 89% and required labor by 46%, while the entire workforce was

successfully retrained for new roles within the company (Al-Rifai, 2024). While the case study was limited to one company, where these methodologies were applied and studied, it shows how the social aspects and stigma that process development leads to downsizing and layoffs can be reversed by both the management and the workforce's resilience and adaptability to retrain within the organization, thereby supporting sustainable organizational growth. Therefore, Al-Rifai's (2024) study contributes to the development of VSM as a socially sustainable tool for process improvement by reducing waste while repurposing the workforce for sustainable growth. This shows how VSM is developing along two paths, firstly by the integration of human factors, as forecasted by Batwara et al. (2023) and evidenced by Al-Rifai (2024), and secondly, how Industry 4.0 will allow the usage of VSM to be tied into the real process in real-time by Reslan et al. (2025).

Altogether, these three concepts of BPMN, Process Mining, and VSM, within the first framework of process analysis and task identification, represent complementary process analysis techniques that are used in industrial settings and how they are developing currently. BPMN establishes the foundation of how business workflows are visually modelled, then Process Mining shows how the workflows are validated by their real performance data and event logs, and lastly VSM shows a Lean methodology for workflow analysis by identifying wasted time and resources. Together, these three concepts provide the analytical basis for this thesis's automation framework by supporting the modelling and identification of tasks that can be securely automated by local AI agents.

2.2 Automation feasibility and levels of automation

After the identification of processes with BPMN, Process Mining, and VSM, their feasibility and level as automation candidates need to be determined. This second framework of automation feasibility and levels of automation reviews studies that define

the characteristics and level of automation applicable for tasks and processes to determine their suitability for automation.

Feasibility in the context of automation answers whether a task or process should be automated. It entails traditional factors such as repetitiveness, rule-based logic, and low variety (Parasuraman, Sheridan, & Wickens, 2000). However, with the newly developed dimension of AI, new automatable groups of tasks emerge that have semi-structured and knowledge-based processes that require perception, learning, or reasoning (Eloundou et al., 2023). These automations can be categorized into different levels based on the extent of human involvement and the system's intelligence, which range from simple rule-based scripts, e.g., Excel Macros, to autonomous AI-driven decision-making systems (Parasuraman et al., 2000). The levels of automation are developed in Parasuraman et al.'s (2000) model, which defines ten levels of decision and action automation. These levels describe how the control of systems differentiates between humans and machines, from complete human control to complete autonomy for machines. The summarized levels are in Table 1.

Table 1. 10 Automation levels of decisions and actions (Parasuraman et al., 2000).

Level	Description of automation level
(High) 10	Decides everything, acts autonomously, ignoring the human
9	Informs the human only if the computer decides to
8	Informs the human only if it is asked to
7	Executes automatically, then necessarily informs the human.
6	Allows the human a restricted time to veto before automatic execution
5	Executes that suggestion if the human approves
4	Suggest one alternative
3	Narrows the selection down to a few
2	The computer offers a complete set of decision/action alternatives.
(Low) 1	The computer offers no assistance: humans must make all decisions and take all actions.

As seen in Table 1, the automation scale progresses from the first level of full human involvement to level ten of complete machine autonomy. With the introduction of AI,

industrial systems are gradually starting to move towards higher levels of automation. Building on the levels of automation in Patrício et al.'s (2025) research on Robotic Process Automation (RPA) and Industry 5.0, their research concludes what characteristics are required for automation of processes using higher-level AI systems. The study was based on a literature review on RPA and Industry 5.0, where the focus has shifted from Industry 4.0's efficiency developments to having sustainable and ethical automation, introducing new requirements for RPA design (Patrício et al., 2025). The research proposes that new characteristics of cognitive tasks are becoming a part of RPA from Industry 5.0 through natural language processing and computer vision, e.g., learning, reasoning, predicting, and adapting to tasks. For the feasibility of these RPAs that have been equipped with AI, Patrício et al. (2025) suggest that the level of the future Industry 5.0 automation will be at a level 5-6, where there are human-in-the-loop solutions in automations, where humans are notified if errors occur and intervene. This is due to the required ethics of how the AI-enhanced RPAs are used, allowing humans to work side by side and provide resiliency to the RPA systems.

Patrício et al.'s (2025) study presents its framework based on a literature review, thus having no empirical evidence of how its automation framework or recognized feasibility characteristics compare in real industrial settings. However, they recognize this as well and state that they plan to conduct empirical research based on this implementation of RPA to Industry 5.0, where they find repeatable processes that require natural language processing, machine vision, or have unstructured data. While Patrício et al.'s (2025) advances integration of AI capabilities into industrial RPA systems, a study by Perez-Cerrolaza et al. (2024) questions the feasibility of high levels of autonomy for RPA enhanced with AI in the use of safety-critical and highly regulated industries, where safe engineering processes are vital. Nonetheless, Patrício et al.'s (2025) research lays the foundations for the new characteristics that are to be considered when determining whether an automation candidate is feasible for automation in Industry 5.0 and outlining an ethical, human-centric adaptation framework.

Building on Patrício et al.'s (2025) research, the development of automation feasibility criteria has also been enhanced by Leng et al. (2024) by examining the new requirements introduced by Industry 5.0. Leng et al. (2024) identified a shift from Industry 4.0's focus on efficiency to Industry 5.0's human-centric and resilient systems, emphasizing ethical and social criteria. Therefore, AI automation is increasingly directed to Level 5-6 in decision and task execution for ethical reasons to ensure employees remain accountable for the work, while letting the AI systems do repetitive and complex operations. The study proposes integrating AI into processes as a cognitive twin or a collaborative robot alongside the human employees, further supporting the argument that Industry 5.0 is developing AI solutions to Levels 5-6 autonomy, where humans remain in charge of process execution (Leng et al., 2024). The research by Leng et al. (2024) defines the vision and scope of future automation feasibility and level, by showing how industrial processes remain ethical while the level of automation needs to be around 5-6, thus outlining that the processes need to be human-centric and resilient for future Industry 5.0. However, like Patrício et al. (2025), Leng et al.'s (2024) research lacks empirical validation and remains a conceptual rather than a tested industrial model. This limitation highlights the current research gap between conceptual visions of Industry 5.0 and their practical validation in industrial environments. Both studies of Patrício et al. (2025) and Leng et al. (2024) pave the way for Industry 5.0's mid-level autonomy, where a balance between efficiency and human accountability is considered.

The levels of automation that Parasuraman et al. (2000) developed were 25 years ago; thus, through technological advancements, the taxonomy and definition of the levels of automation should have changed. Richardson et al. (2025) underline the difficulties with old models and their misuse, while proposing how future models should be developed with their own specialized domains. Autonomous machines were often misclassified, due to the limitations of levels of automation models such as Parasuraman et al.'s (2000), where there are main and subfunctions within the machine, e.g., autonomous cars with cruise control and air conditioning. These main and subfunctions may operate at different levels with each other, leading to inconsistency, where the machine's level of

automation is misclassified, resulting in potential miscommunication in industrial applications (Richardson et al., 2025). The review proposed specialized domains to create their own levels of autonomy by including environmental, regulatory, and human-centric constraints (Richardson et al., 2025). This would then benefit the value of what levels of automation have for each sector, e.g., critical energy infrastructure, aviation, or defense, that allow them to create and control the requirements for levels of automation within their field, thereby enhancing the model's applicability as the intersectoral models would become standardized within their industries. This critique supports the development of automation feasibility frameworks, as it suggests that future models must be domain contextualized. The study by Richardson et al. (2025) was conducted with peer-reviewed articles; therefore, a limitation was that non-peer-reviewed articles were not included, where these intersectoral models for specific levels of automation could have been developed. The findings of improving the models to include sustainability and ethical levels within industries align with the transition to Industry 5.0 that was discussed by Patrício et al. (2025) and Leng et al. (2024). However, the review outlines the misuse and inaccuracies that old levels of automation models have, while proposing new industry-specific models, which improve standardization and create explicit descriptions for the models.

The studies explored in the second framework demonstrate the evolution from descriptive general models and efficiency-focused feasibility to domain-specific models and ethical human-centric feasibility dimensions. The foundational levels of automation taxonomy presented by Parasuraman et al. (2000) have been critiqued by Richardson et al. (2025), emphasizing the need to transition to multilevel and domain-specific models to avoid the current issues of misclassification and miscommunication. In comparison, Patrício et al. (2025) and Leng et al. (2024) highlight that the future automation feasibility depends on an ethical human-centric approach to keep processes safe and resilient. This outlines the level of automation to reach 5-6 on the general Parasuraman et al.'s (2000) model. Though these studies and models remain mostly conceptual, they reveal a gap for empirical validation of automation feasibility in industrial contexts.

From the reviewed studies another clash arises. While in Patrício et al. (2025) study arguments for mid-level autonomy are made, which is ethically motivated from Industry 5.0 values. However, Richardson et al. (2025) shows how generalized automation models, in highly regulated domains, e.g. energy infrastructure, routinely misclassify automation systems. This contradiction outlines the need for specific automation criteria to be developed before adoption. Thus, in this paper's feasibility evaluation for RTU configuration must be based in specific industrial operational risk and procedural constraints rather than in theoretical autonomy levels.

2.3 AI fundamentals: local vs cloud-based AI agents

This framework aims to establish the technological foundation behind AI agents and discuss the benefits and disadvantages of local and cloud-based hosting options with a focus on security and privacy. The current definition for AI agents is that they are autonomous software entities created for goal-directed task execution, while acting in constricted digital environments (Bandi et al., 2025). Their specialty compared to traditional scripts and programs is that they can perceive structured and unstructured inputs and take actions towards specific objectives. They can interpret reactive intelligence by adapting dynamically to inputs and thus reconfiguring their own outputs (Sapkota et al., 2026). This flexibility and adaptability to their inputs and outputs are what drive their adaptation to industry as process and task automation solutions (Sapkota et al., 2026).

To show the operational principles of AI agents, Figure 3 illustrates the fundamental architecture of an AI agent, adapted from Bandi et al. (2025). When the system's hardware and operations remain entirely at a company's own premises, then the agent is considered local. The agent's core component is its reasoning model, which gathers inputs from perceptions and world modelling, e.g., human language text or sensors, and then plans, reasons, and determines based on the input and the agent's memory of how

to reach its goal. After the agent has reasoned and determined its actions, it will use the tools it has, e.g., a chat interface or a command line terminal tool, to execute the actions. Then the agent perceives results based on its actions and reflects on them; if it finds new adaptations, it updates its context in memory. Then the loop will continue until the goal of the agent is met. This closed-loop design allows the agents to have a mid-level degree of autonomy, enabling autonomous problem-solving towards their goals, where human oversight remains possible.

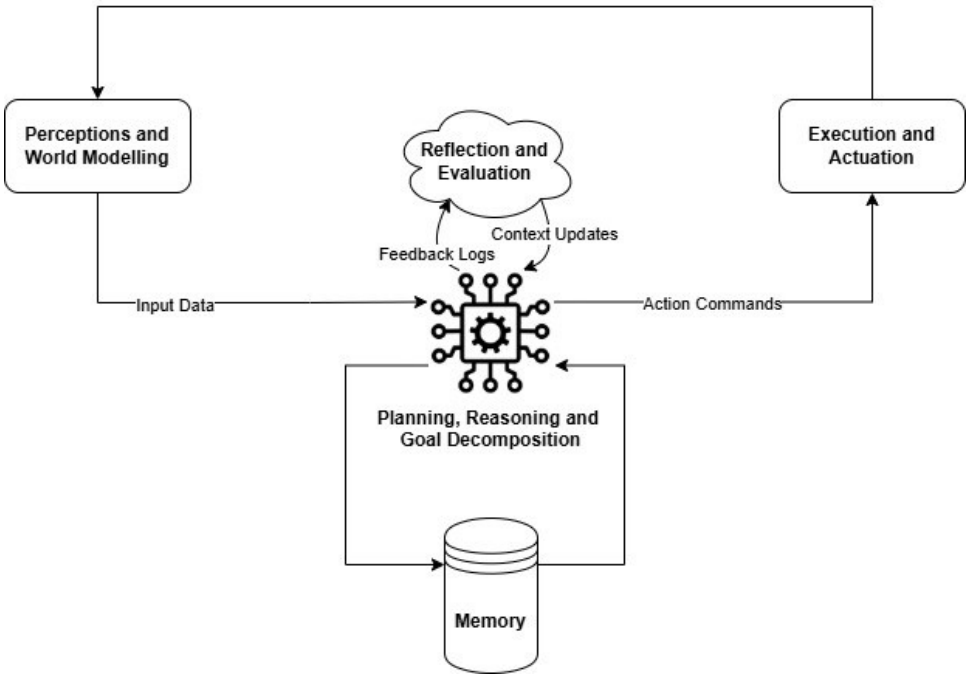

Figure 3. Fundamental architecture of an AI agent.

2.3.1 Key technologies for local industrial agents

To enable the development of industrial AI agents locally, the current capabilities of hardware need to be considered, which drive the adaptation towards Retrieval Augmented Generation (RAG) for vector memory, fine-tuning to specify the model for its tasks, and Small Language Models (SLMs) for its reasoning and goal decomposition component. RAG is a method of giving access to large databases for the AI agent that are not included in its training data. Thus, RAG allows for a larger context for the agent to be used and retrieved using vectors made from the input query that are matched to vectors

of the database and retrieved (Klesel & Wittmann, 2025). Complementing RAG, fine-tuning allows a small number of the model's parameters to be trained to define its output for specific tasks (Klesel & Wittmann, 2025). SLMs are reasoning and goal decomposition components for AI agents that are smaller than Large Language Models (LLMs), allowing them to run on lower-level hardware faster than LLMs (Zhang et al., 2025). The faster and local response proves beneficial for industrial systems requiring low latency and mitigating the need to transmit sensitive data across networks (Zhang et al., 2025). SLMs are created by compressing them from LLMs by quantization to optimize memory usage, pruning to remove unessential weights and neurons, or knowledge distillation, where an SLM is trained to replicate the answers of an LLM, resulting in a black box model (Zhang et al., 2025). Due to the privacy benefits and low latency, SLMs prove viable for industrial AI agents.

Connecting SLMs to industry as a viable option to LLMs, a proof-of-concept study by Griewing et al. (2024) on breast cancer detection supports that SLMs reach the same or required accuracy as their larger counterparts. For clinical oncology, a local Griewing et al. (2024) designed an AI agent that was able to use a fine-tuned local SLM and OpenAI's LLMs GPT4.0 and GPT3.5 as a cross-reference. The SLM achieved treatment recommendation accuracy of 86.0% compared to the GPT4.0 (90.0%) and GPT3.5 (83.0%) (Griewing et al., 2024). With their SLM model, they prove that the AI's transparency and traceability in its decisions by designing the pathways, outlining the link to Industry 5.0's ethical need for automation models. Running the SLM agent on local servers mitigates previous data security barriers, thereby pushing the adaptation of SLM-powered agents for medical organizations (Griewing et al., 2024). The study is highly relevant, as it directly addresses the research gap with the barriers of AI adaptation into the medical industry due to data security and explainability of the agent. A key limitation of the study is that it was conducted as a preclinical proof-of-concept, which does not validate the findings, as they were done on a small fictional sample size of 20 patients. When looking at specific concordances, one out of the five categories, genetic counseling, was the lowest performing for the SLM, where GPT4.0 scored 85.0% and it only got 65.0%

outlining that the SLM clearly underperformed in that category. This shows a limitation of the SLM, but fine-tuning could improve its future performance. Ultimately, Griewing et al.'s (2024) research shows how SLMs can be adapted to industries reaching the same levels of performance, while overcoming data security and decision explainability, which are often cited as goals in Industry 5.0 (Patrício et al., 2025).

Continuing Griewing et al.'s (2024) research on SLMs and their performance compared to LLMs, Zupan's (2025) research on SLM specialization outlines the importance of specialized models that focus their expertise on a narrow domain instead of broad general knowledge. Zupan's study developed a bookkeeping assistant model that knew the case company's accounting principles and history through fine-tuning. Fine-tuning a local model to become specialized was much more successful than trying to use a general LLM, as they tend to be general (Zupan, 2025). Zupan's (2025) research fine-tuned their SLM to understand 17 years of historical company bookkeeping data and processes. The fine-tuning was done using the Low-Rank Adaptation technique, which uses a small number of the SLM's parameters in training, to decrease the computational requirements in training. Hence, by using a local SLM to specialize it for specified sets of tasks by fine-tuning, it becomes capable of executing them with shorter queries, thus making the process time faster and the memory required by the model smaller. A limitation of the full potential of the bookkeeping agent in the study is that RAG was not linked to the agent, but it was discussed by Zupan (2025), who pointed out what to consider when adding RAG for the agent. Adding RAG for the AI agent is beneficial when there is dynamic data that is updated regularly, e.g., company guidelines every quarter, as removing and adding data from RAG's vector database is a dynamic and flexible process. On the other hand, fine-tuning is best for data and processes that always stay the same, as changing the SLM's parameter weights requires repeating the supervised and resource-intensive fine-tuning process (Zupan, 2025). Zupan's (2025) research brings out another successful case study of local AI agent implementation into a bookkeeping process, where the agent is specialized for its function through fine-tuning.

To continue where the key technologies for local industrial agents are headed in Wang et al.'s (2025) study, they project another future application of synergy between SLMs and LLMs to use RAG. Wang et al. (2025) research describes a RAG framework where SLMs decompose a query, LLMs perform reasoning from the retrieved RAG data, and SLMs evaluate the outputs, hence creating a transparent RAG pipeline that enhances the accuracy of the answers and their explainability. This identified RAG framework was successfully used to improve the answers of an AI agent that received queries for industrial Telecom Networks (Gichamba et al., 2024). Wang et al.'s (2025) study suggests further future implications of specialized SLMs in hybrid work with LLMs, which could oversee and safeguard the LLM's performance by reviewing the LLM's answers, thus providing a more secure AI agent. While the proposed RAG framework from Wang et al. (2025) improves answer accuracy, its implications in industry for the use of cross-models would raise the need for computational power and data security when transferring data between the SLM and LLM. Nonetheless, Wang et al.'s (2025) research on the coworking models between SLMs and LLMs implies a future where they work in unison to provide safer and accurate systems.

2.3.2 Secure on-premises vs cloud-based deployment

To understand why an on-premises AI agent can be considered more secure, the current literature on local AI and cloud-based AI is reviewed. An important concept for on-premises is edge deployment, which is a paradigm shift for the industry where machine learning models are placed at the edge of the network, where the AI is moved from cloud services to the local devices on the local network (Meuser et al., 2024; Singh & Gill, 2023). From Figure 4, the locality of edge AI can be observed compared to cloud AI, which operates in a remote location through centralized datacenters. Thus, edge deployment of AI shows potential in being a low-latency and secure option for industrial automation due to its data security factors, while having its challenges with computational power and energy efficiency.

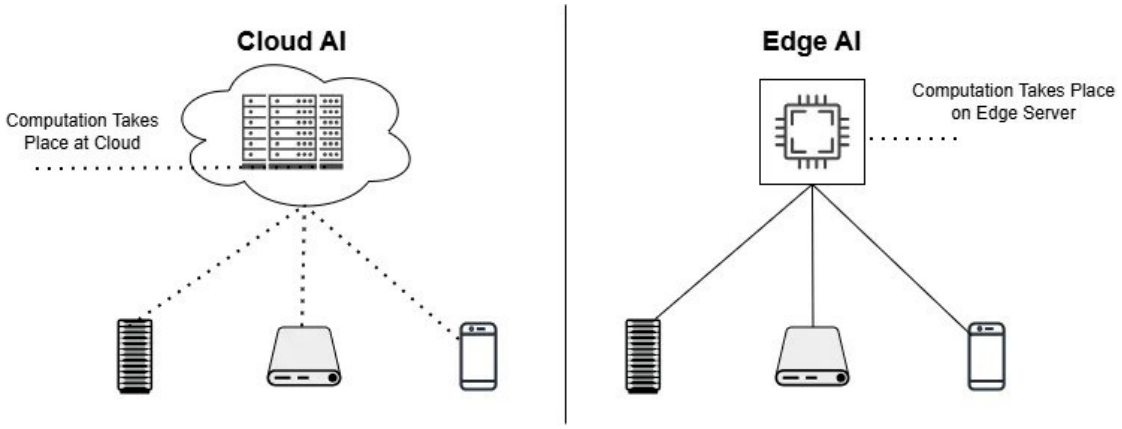

Figure 4. Cloud and edge AI's computational place.

Lower latency, improved privacy, and better responsiveness are proposed through edge AI as a paradigm shift happening to industrial AI automation in research by Meuser et al. (2024). Their research concluded that, along with the historical evolution of cloud computing to edge computing, the integration of AI is following. Integrating AI into industrial processes that require low latency by using edge AI solutions proves to be valuable, as compared to cloud AI, which may suffer from possible network latencies or outages. The benefit of cloud AI is the computational advantage and scalability, as edge AI requires the company supplying it to procure the hardware and build the edge AI solution, while cloud AI can be procured as a service from a datacenter, e.g., Google (Meuser et al., 2024).

Their research brings up data privacy and GDPR concerns of cloud-based solutions, which outline the current state of cybersecurity in companies to keep sensitive industrial data on-premises. Although the successful implementation of edge AI depends on the limitations of hardware constraints, integrating perception into the AI agents and models through sensors successfully (Meuser et al., 2024). On the other hand, this proves an advantage for companies engaging in such AI adaptation to allow the deployment of autonomous agents into industrial settings. Meuser et al.'s (2024) research in creating specialized SLMs for healthcare diagnostics, autonomous driving, and industrial automation is validated empirically by Choudhary et al. (2025), who study the performance of edge and cloud-deployed AI. Their edge deployment reduced the SLM's

latency by 45%, from 120 to 40 seconds, improved accuracy from 85% to 94%, and lowered energy use from 5.2 W to 1.5 W compared to cloud-based models. Therefore, Meuser et al.'s (2024) study, which is complemented by Choudhary et al.'s (2025) empirical proof, describes a path of industrial automation with autonomous AI agents deployed on the edge for more time-sensitive and privacy-concerning applications.

Fraga-Lamas et al. (2021) demonstrate how on-premises AI solutions promote sustainable AI, which builds on the work and findings from Choudhary et al. (2025). This perspective expands the width from privacy and latency to sustainability and energy efficiency. Local on-premises AI is more sustainable, even though cloud solutions offer unlimited scalability, they do require continuous and high-bandwidth transmission of raw data to distant datacenters, e.g., video and high-frequency sensor readings (Fraga-Lamas et al., 2021). Their study proposes a green edge AI solution where data transmission is minimized. Sustainable edge AI is therefore a solution and a goal for the development of AI solutions in industry, reducing the need for high rates of data transmission (Fraga-Lamas et al., 2021). Meuser et al.'s (2024) highlights this by the introduction of implementing local SLMs, which allow process automation on-premises, lowering their energy consumption. Furthermore, the use of local AI agents directly supports the strategic Industry 5.0 goals, with its definition of being the shift from efficiency-oriented processes to more sustainable and human-centric processes. Thus, Fraga-Lamas et al.'s (2021) study shows how industrial automation through local edge AI deployment demonstrates that it will play a part in achieving the goals of Industry 5.0.

To conclude the third framework of AI Fundamentals, the literature reveals a clear gap in the industry for specialized SLMs that are trained depending on the task with fine-tuning and RAG. This training allows the AI agents to be more energy efficient and focused on the task they are assigned for, driving Industry 5.0's strategic goal of sustainability. The challenge of sensitive data and GDPR compliance is addressed with edge AI, as there is no need to send vulnerable organizational data to the cloud's datacenters. However, while the implementation of local AI reduces data security risks,

it also transfers responsibility for cybersecurity oversight and system maintenance responsibilities to the organization, which may limit the adoption of local AI agents in smaller companies with fewer resources.

3 Research methodology

The research on secure industrial process automation will follow a single-case Design Science methodology with problem identification, design, evaluation, and reflection, where the reflection stage evaluates the feasibility criteria (Schultz, 2017). The approach is centered on a local AI agent prototype for RTU configuration as an artifact. Design Science was chosen over modelling and action research, since the primary goal was developing and evaluating the artifact, rather than modelling and observing occurring phenomena. Additionally, with the construction and evaluation, Design Science supports research questions 1 and 3, in terms of the feasibility and performance of the AI agent. In the study, both qualitative and quantitative measures are considered with the quantitative performance of the artifact, and qualitative measures of the process through Process Mining and VSM, resulting in a mixed-methods approach. Thus, the study's methodology is single-case Design Science, although being limited in broad generalization due to the results being context-bound, proving to be the most viable for such artifact development.

Alternative methodologies were evaluated but concluded to be less suitable. Action Research was rejected as the objective of the study was to prototype the artifact and determine its feasibility, and not to drive organizational change with e.g. AI adoption. A multi-case comparative design was another option, however due to the comparable RTU environments not being accessible as of data security concerns between different organizations. These constraints, implied by the other methodologies, limit the thesis's external validity, however they show strength in better focus for the single artifact's measurements and repeatable workflows. Hence, the selected Design Science approach aligns as the most optimal methodology, with the research objectives of feasibility and safety-critical configuration.

3.1 Case description

The case study was conducted at Hitachi Energy's Grid Automation unit in Vaasa, where the RTU configuration process is the unit of analysis in the study. RTU configuration was chosen due to its repetitiveness, rule-based, and safety-critical workflow, which consists of measurable outcome indicators. The measurable success indicators and standard processes allow controlled performance benchmarking, e.g., executing and retrieving backups. While the artifact developed in this study only focuses on Hitachi's command-line configuration syntax, the study acts as proof of concept for similar configuration processes in SCADA environments for other command-line configurable devices.

The measured configuration process consists of three subprocesses: file uploading and updating, user credential management, and backup generation and restoration. For each of the tests, these subprocesses will be executed consecutively as a full workflow. Moreover, the physical test setup for the case study will consist of three Hitachi RTUs, of which two are RTU530 models and one is an RTU560 model. They are connected through a switch to a Local Area Network (LAN), with an engineer's laptop where Process Mining and the local AI agent are hosted. The latest version of firmware is used, and each RTU has its specific configuration files created with the RTUtil tool. The data collected from testing will be anonymized for any sensitive data and labelled according to RTU560, RTU530A, RTU530B, and the engineer's laptop. Data security is also addressed, as the testing happens on-premises at a private LAN, with no data leaving company systems. The case environment provides repeatability and security for the artifact validation.

3.2 Data collection methods

The data collection of the study was divided into three stages. Firstly, process capture and analysis, secondly, dataset creation and model fine-tuning, and lastly, artifact testing. Hence, with the three data collection stages, the study's mixed-methodologies are

applied for quantitative empirical data's performance evaluation and qualitative modelling of the analysis process feasibility.

3.2.1 Process capture and analysis

Process Mining logs are captured during manual runs of RTU configuration to identify the three workflow subprocess flows and cycle times. The logs are captured with Microsoft's Power Automate Process Mining, which observes desktop activities. The following event logs and process maps will be used to build a Value Stream Map to further analyze the processes' non-value-adding steps. The combination of Process Mining and VSM enables quantitative analysis with time and qualitative waste identification, thus providing the research with automation feasibility evaluation criteria.

3.2.2 Dataset and artifact creation

The supervised fine-tuning was executed on Google Colab's server with an Nvidia T4 GPU, using the Unsloth framework for fine-tuning. Due to Colab's 4.5-hour continuous session limit, the training duration was designed to last an hour to allow dataset preparation and model initialization.

This session limitation was met with the dataset of 400 prompt-response pairs per subprocess, trained for three epochs. Each pair was a configuration example of a user query with randomized parameters for IP addresses, user credentials, and file paths, and a structured response from the model with reasoning and PowerShell commands. Therefore, the dataset was optimized for training data, and the duration of training that could be achieved with the hardware session constraints.

Several open-source SLM providers were considered for artifact feasibility, but Alibaba's Qwen2.5-coder models were chosen due to their high performance in code generation tasks (Hui et al., 2024). Due to the engineer's laptop's hardware constraints of 4GB of RAM in its Nvidia T1200 GPU, only models with parameters below 4B could be deployed

locally, with a feasible token generation speed. During pilot testing out of the 0.5B, 1.5B, and 3B models, the 1.5B parameter model was as accurate and faster than the 3B model. Therefore, qwen2.5-coder:1.5b model was chosen due to code dataset training, its small size, and the possibility for structured output, which was necessary for the artifact.

The artifact was developed as a local web application that runs the SLM through Ollama at its backend. The artifact takes in user queries with few-shot examples of RTU configuration and returns structured output with reasoning to the user and PowerShell scripts, which are executed by the terminal tool in the backend, which connects to the RTUs. This artifact architecture ensures that executions stay on-premises for data security.

3.2.3 Testing procedure

The testing was conducted twenty times for each of the subprocesses, using the base model, the fine-tuned model, and having a manual engineer benchmark. All tests log execution time, success rates, errors, and system metrics for the GPU's power usage. Errors occurring were categorized into syntax, file path, and command errors. Additionally, qualitative notes are collected on model hallucinations and workflow observations.

3.3 Data analysis and metrics

The data analysis combines quantitative performance metrics and qualitative process evaluation to measure the artefact's technical and operational feasibility. Quantitative analysis is focused on the test variables in Table 2, which include completion time, success rate, and estimated power consumption. Paired t-testing is conducted between the manual and AI agent runs with a 95% confidence level to determine the differences in task completion time and error rate. Cohen's d is used to determine the scale of effect of using an AI agent for the different subprocesses. The GPU usage is logged to estimate

the power consumption that applying such agents would have compared against cloud-based LLMs for sustainability metrics. Therefore, with quantitative variables, the performance and feasibility of the AI agent are evaluated holistically through pure efficiency, while also considering its sustainability.

Table 2. Quantitative and qualitative variables were used in the study.

Quantitative	**Qualitative**
Completion time (s)	Error types
Success rate (%)	System behavior
Estimated power consumption (W)	Consistency across runs

For qualitative analysis, the error types and number of errors were recorded, with the overall system behavior, and Process Mining and VSM diagrams will be developed. Thus, Process Mining and VSM diagrams will be used to interpret tasks where efficiency gains occur and to help analyze the processes for their characteristics that make them suitable for local AI agent automation.

3.4 Reliability and ethics

For the study's reliability, all the tests are conducted in the test laboratory, with identical firmware, consistent configuration files, and network setup. Additionally, an identical task sequence and operation ensure reliability in the testing phase. Although the case study is limited to Hitachi's RTUs, the findings can be extended across similar industrial configuration processes that rely on internal and company-specific scripting syntax and documentation. The study is aligned with the ethical guidelines, as confidential data is kept on-premises and the event log data is anonymized, hence the testing follows Hitachi's IT guidelines and GDPR regulations. Thus, the combination of process mining, benchmarking, and transparent reporting supports replicability and academic validity.

4 Results and findings

The artifact proved to be a successful tool in allowing a human operator to use natural language to configure Hitachi's RTUs. All subprocesses were executed and measured with differing artifact capabilities, due to perception restrictions and RTU software limitations. The file upload and updating subprocess resulted in being the most successful, with all measurements completed. The next subprocess of user credential management yielded no results due to the AI agent being incapable of providing multistep instructions without receiving feedback from the RTU. The RTU backup process was successful, though being limited to only RTU530-series, as RTU560 does not support backup usage. The runs made on the fine-tuned SLM were able to do the same tasks, as shown in Figure 5.

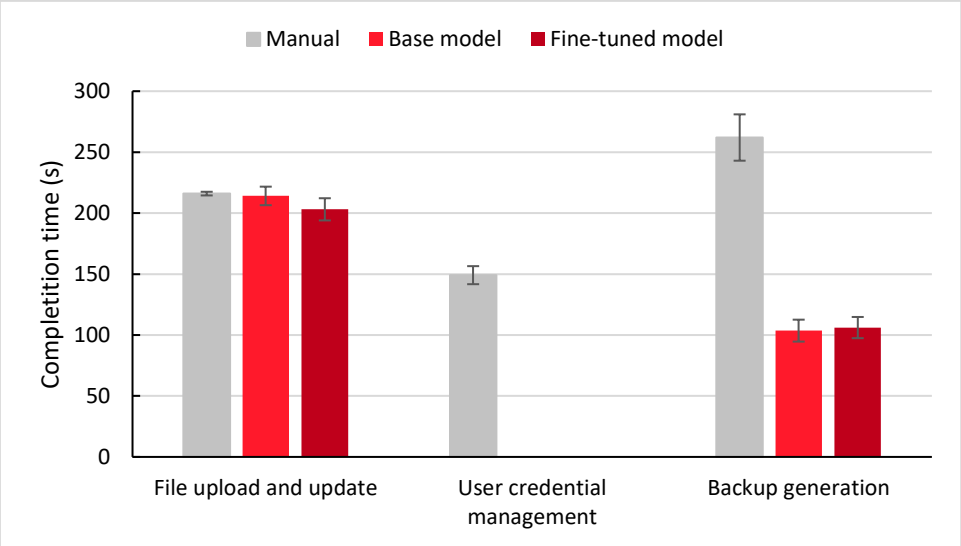

Figure 5. Subprocess execution performance in mean time with standard deviation.

The artifact showed level five automation, where it creates the workflow depending on the user query and allows the user to approve and execute it. Therefore, the artifact showed proof of concept in applying an internal scripting syntax for workflow processes through a locally deployed AI agent.

4.1 Process Mining and Value Stream Analysis

Process Mining resulted in 30 captured runs, divided into 10 captures, which were captured consecutively. These captures were analyzed with Power Automate and resulted in Figure 6 and Figure 7 Process Mining maps. From the heat map, the most frequent steps are visualized through flow arrows and explanatory stages, which were logging in to the configuration user interface, file searching, and navigation through the user interface.

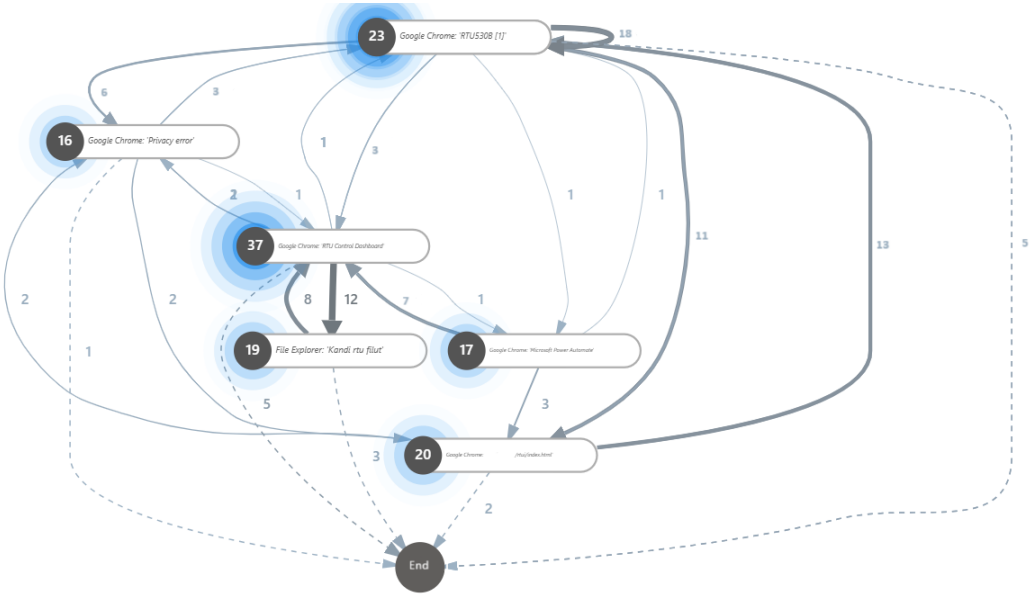

Figure 6. Process Mining map of webpage visits during configuration tests.

However, Power Automate did not suggest any automations for the process from the captured tasks after its analysis. The average time spent on different webpages is visualized in Figure 7. Thus, showing that most time was spent on the configuration user interfaces for different RTUs between 1-2 minutes, while time spent on the file explorer averaged 13 seconds, thus most of the configuration time was spent on each RTU's webpage user interface for configuration.

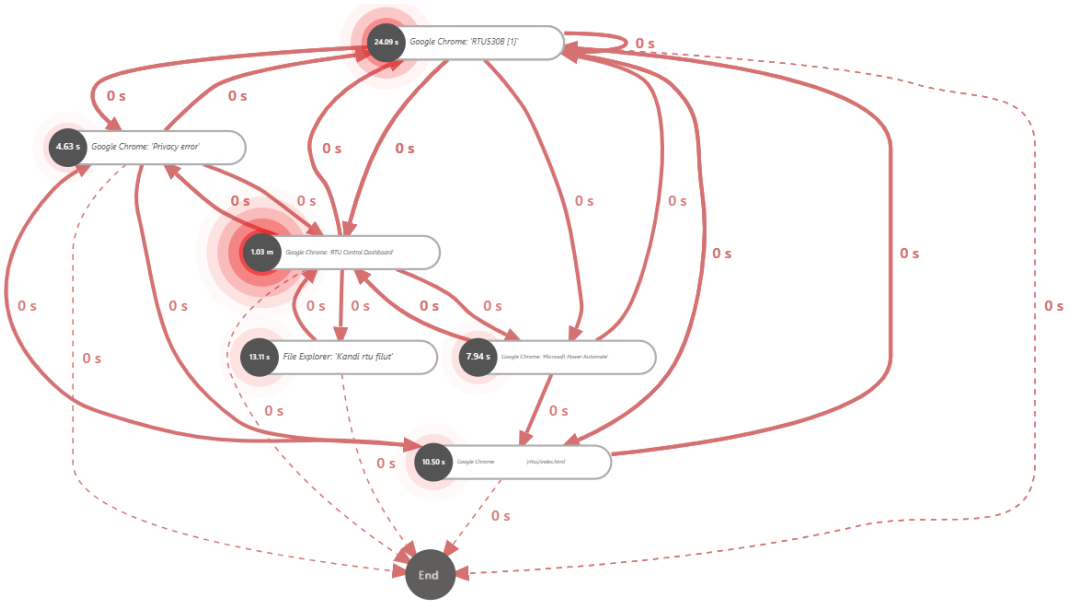

Figure 7. Process Mining heat map of time spent on webpages.

These Process Mining results do not themselves facilitate the creation of the Value Stream Map for the configuration process; however, they show a direction and give estimated flows, which can be applied by the process analysts for the VSM. The results of both Figures 6 and 7 are taken into consideration with the results gathered from the testing and the test engineer's knowledge to develop a VSM of the RTU configuration process.

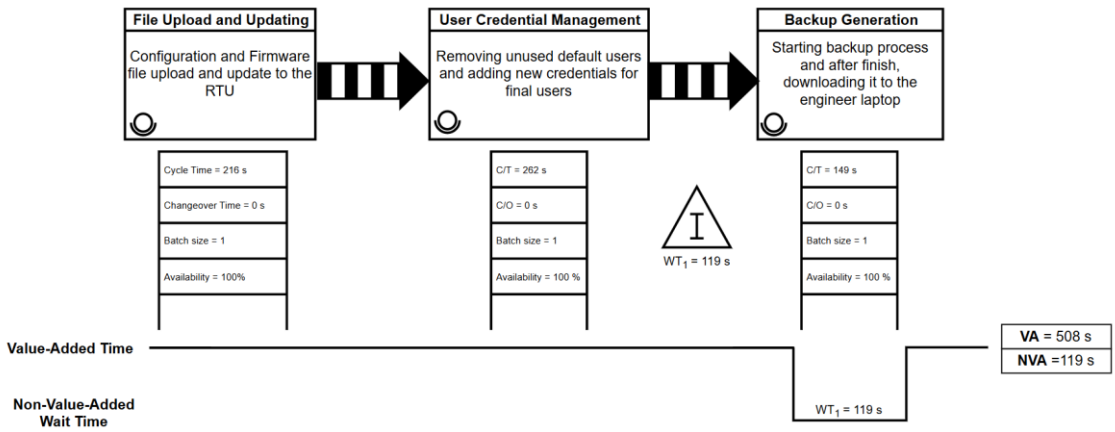

Figure 8. Value Stream Map of the test configuration process.

The developed VSM in Figure 8 shows that the whole manual configuration process took on average 627 seconds for all three RTUs, where 119 seconds were labelled as the wait time, as for each of the two RTU530s from the Process Mining event logs, a total of 119 seconds were spent idle waiting for the RTUs to generate backups. The VSM of the configuration process as a whole allows for full visualization of the testing process and its flow. The cycle times are the mean times taken from the quantitative results of manual subprocesses, and as the tasks were done one RTU at a time, the batch size and availability are one and 100% respectively. The change over time for testing remains at zero seconds due to the testing being fulfilled at the same location and the RTUs being connected to each other.

4.2 Quantitative findings

Results were logged in each subprocess and analyzed for their performance. Table 3 shows the file uploading and updating performance, where the success rate was 100%. The manual and the AI agent modes did not show substantial improvements, as the Cohen’s d value for the base model was 0.25 and for the fine-tuned model 0.06. The t-test shows that due to the p-value being higher than 0.05 for both the base and fine-tuned model, there is no statistical significance in the improvement of the task's speed. For the base model, the confidence interval of 95% was between -1.7 and 5.5 seconds, thus containing zero in the range and supporting that there is no statistically significant effect on the file uploading and updating tasks' time. Even though the fine-tuned model is on average 12.9 seconds faster than the manual method, their tests showed a big confidence interval between -86.4 and 112.3 seconds; hence, with the high p-value resulting in no statistically significant improvement or decrease in the configuration tasks' time. However, the file upload and updating process showed a 100% success rate, while the time taken for an engineer to use the AI agent for configuration did not statistically enhance performance throughput time.

Table 3. File upload and update performance.

Test type	Mean time (s)	Success Rate (%)	Cohen's d	p-value
Manual	216.0 ± 1.5	100	-	-
Base model	214.1 ± 7.6	100	0.25	0.28
Fine-tuned model	203.1 ± 9.1	100	0.06	0.79

Note: For Tables 3, 4, and 5, manual test types Cohen's d and p-value are left blank due to being the comparison baseline; these values outline the statistical significance and effect size relative to manual performance.

In user credential management the artifact was not successful in completing the same workflows as the human engineer was, which can be seen from Table 4. Successful user credential management required multiple steps and feedback to the AI agent from the PowerShell terminal, which was not supported, thus resulting in the creation of workflows, which were half of the required or had contained placeholders, resulting in syntax errors.

Table 4. User credential management performance.

Test type	Mean time (s)	Success Rate (%)	Cohen's d	p-value
Manual	149.1 ± 7.4	100	-	-
Base model	-	0	-	-
Fine-tuned model	-	0	-	-

Backup generation was a very successful subprocess for the AI agent due to the possibility of running the task in parallel among multiple RTUs. This comparison shows the difference between the manual backup of two RTU530s with a success rate of 100% among all testing types. The tasks' throughput time was significantly improved, which could be seen from Table 5, Cohen's d being 9.34 for the AI agent with the base model and 7.23 for the fine-tuned one. Analyzing the base model with its t-test result of a p-value of less than 0.001 proves that the result is statistically significant and there is a difference in the task competition time. From the base models' 95% confidence interval,

which is between 150.5 and 166.4 seconds, as both the upper and lower boundaries are positive, the AI agent is significantly faster than the manual backup process. For the fine-tuned model, the upper and lower boundaries of the confidence interval were 145.9 to 166.1 seconds faster than the manual backup method, portraying the same improvement in the task's throughput time.

Table 5. Backup generation performance.

Test type	Mean time (s)	Success Rate (%)	Cohen's d	p-value
Manual	262.0 ± 19.0	100	-	-
Base model	103.6 ± 9.0	100	9.34	0.00001***
Fine-tuned model	106.1 ± 8.7	100	7.23	0.00001***

*Note: *** p<0.001.*

The usage of power was measured over 100 queries with an even split from the base model and the fine-tuned model. The resulting GPU processing duration of 5.0 seconds was reached from both datasets; thus, their results were combined into Table 6. The resulting figure of 0.037 Wh per query was found to be the average energy usage with the AI agent.

Table 6. GPU power consumption per user query.

Duration (s)	GPU power (W)	GPU idle power (W)	Energy per query (Wh)
5.0	30.6	4.4	0.037

Quantitative errors in the testing were mostly from the user credential management limitations of the agent, resulting in 20 command errors for both the base and the fine-tuned model. The fine-tuned model did not experience any other errors, while the base model had one command error in both file uploading and backup generation. The file uploading error resulted from the model trying to use PowerShell variables in the generated script. The backup command error came from the model trying to fetch available backups with PowerShell commands. Therefore, the base model resulted in 22 command errors, and the fine-tuned model resulted in only 20 command errors.

4.3 Qualitative findings

For qualitative findings, the major areas under observation were errors, system behavior, and consistency across runs. In addition to these areas, the results from Process Mining and VSM were developed for the process analysis.

Qualitative outcomes of the errors, in addition to the ones discussed in the quantitative errors, were syntax errors from the user. The user engineer in testing had three runs where the prompt had a spelling error, but it did not affect the output and result of the workflow, as the spelling error did not occur in the file names or the file paths.

4.4 Fine-tuning results

RTU AI agent's fine-tuned model took a total of 381 steps and 22.2 minutes for training. The final training loss at 350 steps was 0.14 over the training data, which can be seen from Figure 8. However, the model's final loss after the 381 steps of fine-tuning was 0.58. The fine-tuned model developed in this thesis is openly available in Hugging Face: https://huggingface.co/Courbeau/qwen-1.5b-rtu.

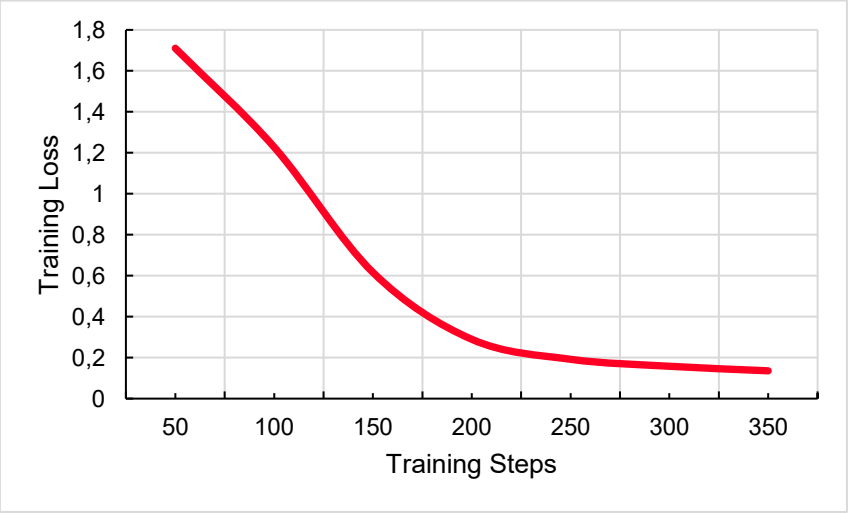

Figure 9. Fine-tuned the model's training loss over training steps.

After fine-tuning the model, it was tested without a system prompt for its responses, and qualitatively, it was observed that the model had learned the desired JSON syntax and some syntax and commands of the RTU scripting interface, which can be seen from Image 1 of the terminal output of the fine-tuned model.

```
>>> Backup RTUA as "BackupA.bakpkg" to path Backup RTU530A as "RTU530abackup.bakpkg" to path "C:\Users\
                    \Desktop\Kandi rtu filut"
reasoning: Generating PowerShell commands to backup the specified RTUs and save them to the specified paths.

powershell:
New-Backup -Name BackupA -RTUAddress RTUA -FilePath "C:\Users\                                  \Desktop\Kandi rtu
filut\BackupA.bakpkg"
Receive-Backup -Name RTU530a -RTUAddress RTU530A -FilePath "C:\Users\
      \Desktop\Kandi rtu filut\RTU530abackup.bakpkg"

total duration:       2.4324855s
load duration:        94.2413ms
prompt eval count:    302 token(s)
prompt eval duration: 146.2121ms
prompt eval rate:     2065.49 tokens/s
eval count:           132 token(s)
eval duration:        2.0661355s
eval rate:            63.89 tokens/s
```

Image 1. Fine-tuned model's terminal output on backup task.

However, the model could not output functional commands without the system prompt with a few-shot examples. From Image 2, the difference in output behavior can be seen through the general output of the base model. The base model does not have knowledge of the correct output commands, as it uses general PowerShell syntax along with explanations. Therefore, the fine-tuned model was successful in learning the required output JSON syntax and shows improvements in Hitachi's internal RTU Scripting Interface command knowledge.

```
>>> Backup RTUA as "BackupA.bakpkg" to path Backup RTU530A as "RTU530abackup.bakpkg" to path "C:\Users\
                    \Desktop\Kandi rtu filut"
To automate the backup process of the RTUA to a specified path, you can use PowerShell. Below is an example script
that will copy the RTUA backup files from one location to another:

```powershell
# Define the source and destination paths
$sourcePath = "\\BackupRTU530A\RTU530A.bakpkg"
$destinationPath = "C:\Users\                                  \Desktop\Kandi rtu filut"

# Ensure the destination directory exists
if (-Not (Test-Path $destinationPath)) {
    New-Item -ItemType Directory -Path $destinationPath
}

# Copy the RTUA backup file
Copy-Item -Path $sourcePath -Destination $destinationPath

# Display a confirmation message
Write-Host "Backup completed successfully. File copied to: $destinationPath"
```

Explanation:
1. **Define Paths**: The script defines the source path (`\\BackupRTU530A\RTU530A.bakpkg`) and the destination
path (`C:\Users\                                  \Desktop\Kandi rtu filut`).
2. **Check Destination Directory**: The script checks if the destination directory exists. If not, it creates the
directory.
3. **Copy File**: The `Copy-Item` cmdlet is used to copy the file from the source path to the destination path.
4. **Confirmation Message**: A confirmation message is displayed to indicate that the backup was completed
successfully.

Usage:
1. Save the script to a file, for example, `BackupRTU.ps1`.
2. Run the script in PowerShell with administrative privileges. You can do this by right-clicking the script and
selecting "Run as administrator".

This script ensures that the RTUA backup files are backed up to the specified path securely and efficiently.

total duration: 6.761261s
load duration: 89.5635ms
prompt eval count: 91 token(s)
prompt eval duration: 555.7693ms
prompt eval rate: 163.74 tokens/s
eval count: 392 token(s)
eval duration: 5.7185564s
eval rate: 68.55 tokens/s
```

Image 2. Base model's terminal output on backup task.

5 Discussion and Conclusion

5.1 Discussion

The performance and feasibility of the RTU configuration AI agent will be evaluated and referenced to the literature in this subsection. Its security and sustainability will likewise be analyzed and linked to real-world practice. Furthermore, the limitations of the study are evaluated, and its implications for follow-up studies are discussed.

5.1.1 Interpretation of results and relation to research questions

The first research question inquired about the characteristics of industrial workflows, which make them feasible for local AI agent automation. The scope of the research was limited to RTU configuration workflows, which are done in factory testing environments mostly digitally through a web user interface, with some manual labor in setting up power and the LAN for the RTUs.

Automation was feasible to an extent, as the subprocesses of file upload and updating, and backup generation were successful when fulfilled with the AI agent, showing feasibility. However, the user credential management subprocess was unsuccessful, which was due to the agent's limits in interacting with the terminal and receiving feedback from it, thus showing a point of improvement in multi-step configuration. Moreover, what made the AI agent automation possible was Hitachi’s RTU scripting interface module, which allowed the use of the engineer's laptop’s PowerShell terminal as a tool for the AI agent.

Furthermore, the module’s own commands and syntax were made possible on a digital interface to automate the configuration process by teaching the local AI agent of its usage. In addition, the creation of model examples by the engineer from the scripting interface datasheet allowed the successful learning of the workflow for the SLM used in the AI agent. Therefore, the characteristics for feasible local AI agent automation include

the possibility for LAN deployment, a tool with which the AI agent can interact with the required task, compile correct examples and data of the tool’s usage and workflow.

The security of local AI agents in comparison to cloud-based solutions proves immensely important, due to keeping sensitive data in the operating organization's network. This was proved by the study, which successfully allowed the AI agent to configure RTUs in a LAN. While the experiments conducted in the thesis did not contain sensitive customer information, real engineering projects would have sensitive customer information such as IP addresses and user credentials. For that customer information, it is exceptionally important to keep it safe, as a data breach would allow malicious attackers to take advantage of RTUs, which control, e.g., the Finnish power grid and cause blackouts. Therefore, the achieved AI agent performance locally shows how edge deployment is a solution for future automation systems, which benefit from AI agent automation, but contain sensitive data in their processes.

The third research question was concerned with the performance of an AI agent in comparison to a human engineer. Firstly, the level of automation that was achieved with the AI agent was level 5, where, based on the human query and task, the AI agent suggested a workflow, which the human had to approve before initiation. This choice is a safe approach regarding the safety of AI systems, as this AI agent had access to the engineer's laptop terminal, thus maleficent through prompt injections and hallucinations are a possibility. On the other hand, as the agent was detached from the network and the commands were coming from a human engineer, the risk for prompt injections through third parties is minimal.

However, the agent was successful in two of the three subprocesses. Backup generation was successful and quicker than the manual operations, as the artifact AI agent was designed to allow parallel sessions to RTUs, thus allowing two backups to be done simultaneously, resulting in Cohen’s d being between seven for the fine-tuned model, which is a substantial improvement. However, it is due to architectural design and not

the SLM. For file upload and updating subprocess results, the agent's and the manual methods completion time did not show statistical significance for improvement, showing that it performed to the level of a human engineer. For the user credential management subprocess, the agent was incapable of fulfilling the subprocess due to its required multi-step interaction with the PowerShell terminal, which was not included in the artifacts design, thus resulting in all tests failing for the subprocess.

Therefore, the failure in user credential management subprocess exposed a structural limitation, showing how local SLMs cannot yet perform fully autonomous terminal interaction without supervised execution control layers. This finding outlines the importance of human-in-the-loop control over critical subprocesses in industrial automation, in order to prevent unsafe command execution and ensure the controlled results. An alternative approach to the finding would be that the limitation does not lie within the model but the architecture choice in the artifact around it isolating its terminal actions behind a predefined tool wrapper. Thus, future versions of the AI agent should evaluate whether which restrictions for the agent's tools can facilitate the automation to move to the next level with e.g. command whitelisting or deterministic policies.

Further findings from the research include the energy consumption of the AI agent and the fine-tuning results. The sustainability of the local SLM is significantly greater than that of an LLM, as the SLM used 0.037 Wh, while GPT-4.5 has a power usage of 6.7 Wh (Jegham et al., 2025). This power usage is for a similarly sized query, as the test query was approximately 200 tokens to the model and 300 tokens out of the model, and in Jegham et al. (2025) research had 100 tokens into the model and 300 tokens out of the model. Hence, a prompt with GPT-4.5 uses 181 times as much energy as the local SLM of qwen2.5-coder:1.5B. Thus, the energy consumption of cloud-based LLMs such as GPT-4.5 is massive in terms of sustainability and energy consumption. However, the energy consumptions of all LLM estimation vary; nevertheless, Jegham et al.'s (2025) study outlines a direction for them. This shows that the future of AI automation cannot

currently be fulfilled with only high-energy-consuming LLMs, as the deployment of more sustainable SLMs is required.

For the fine-tuning results, a final loss of 0.58 was reached after training, which is high for specific syntax generation tasks, implying that more training should be done. However, this proved to be sufficient to recognize changes in the SLM behavior, as without giving the model a structured prompt, it was able to output the required format and try to output the proper commands. Hence, fine-tuning the model was successful, but it did not show any improvement in the query times, due to the queries being short, and the examples of the base model were sufficient for the operations. The speed of the base and fine-tuned model would be evident if the base model's system prompt contained hundreds of examples of different scenarios, while the fine-tuned model would be trained on the examples and would have a short system prompt; then the speed advantage of fine-tuning would be visible. In addition, the fine-tuned model had fewer errors than the base model, proving that the training enhanced its capability to produce the required commands, syntax, and structure. Thus, fine-tuning in this case study did not provide enhanced efficiency; rather, it improved the agent's consistency.

5.1.2 Comparison with existing literature and practical implications

Local AI agents through SLMs can execute workflows with similar performance to cloud-based LLMs and do it sustainably. Griewing et al. (2024) study on implementing an SLM for breast cancer detection treatment performed to the same metrics as a cloud-based LLM did, thus complementing this research in successfully deploying local SLMs for industrial workflows.

Supporting the movement of Industry 5.0 with sustainable and efficient edge deployment is supported by Alzu'bi et al. (2025). In their study, they noticed a 30% energy usage reduction between their cloud-based deployment and edge deployment of AI to their Internet of Things devices. Thus, supporting the sustainable change of Industry 5.0 with edge AI deployment to industrial environments.

Studies explicitly do not discuss the challenges of autonomous multi-step iteration for AI agents, allowing them to interact with tools themselves over multiple times. However, the human-in-the-loop approach is used across many studies in academia, supporting the conclusion of creating agents of automation level 5, where a human operator approves the workflow proposed by the agent. An example of this was in a pedagogical AI agent, which had multiple gates between its lesson overview, storyline, and scoring, supporting level 5 automation (Chow & Ng, 2025). Therefore, the challenge of accurate and autonomous multi-step actions from AI agents is uncovered based on this study.

Moving to the practical implications of the study for engineers, project work, and real RTU configuration, which showed advantages and faults within the artifact developed. The artifact proved to be more sustainable with lower GPU power consumption compared to cloud-based LLMs. It succeeded in keeping sensitive data on-site by edge deployment and local inference, and the design architecture of managing multiple RTUs at once allowed for parallel workflow execution, enhancing the throughput of the configuration process. However, it failed in credential management, as it was a subprocess requiring the agent to use the terminal to gather information about the users and then produce its workflow, hence showing where industrial AI agent adoption for RTU configuring is blocked. Therefore, the study implies that an RTU AI agent is feasible to develop locally and securely if its multistep workflows are revised.

5.1.3 Limitations and potential follow-up studies

The technical limitations of the study constrained it and were visible in the results in the user credential management failures. This was due to the limitation of an interactive loop with PowerShell, which the AI agent did not have. Another limitation, which affected the results, was that the RTU560-series does not support backup generation, as only RTU530 devices could be tested with the backup subprocess. Furthermore, the study was conducted in a single-case setting, which hinders the generalizability of the results. From the fine-tuning results, the model learned the responding syntax, while it

did not learn the exact commands of the RTU scripting interface, proving that by itself, the fine-tuned model would be limited and would not function. Overall, these technical limitations point towards how the study was a single case study, device capabilities within the RTU500-series, and limitations with the AI agent and its fine-tuned SLM.

A further limitation of the study being a single case study with only one engineer testing the test setup plays a part. The results might have largely been affected by a systematic error due to the operator skill in question, as the usage of the configuring webpage and AI agent differ with different engineers, thus resulting in over- or underestimation of the agent's relative performance and stability. A test environment with multiple engineers from Hitachi's project team would strengthen the external validity of the results. Nevertheless, the controlled setup was necessary due to resource constraints.

The limitations of the artifact in terms of its being applicable to use in real project work at Hitachi would be integration of autonomous access to the PowerShell terminal, the integration of project files into the user interface of the artifact, and an RTU lookup tool. An additional tool that the AI agent would benefit from is a tool for browsing user files and folders for project files. Solving this limitation would improve the usability of the agent, as for the artifact in the study, the engineer's time was spent searching for the right file paths from the file explorer and setting them. Thus, the artifact would benefit from a project folder navigator for both the engineers using it and as a tool for the AI agent to use.

The discussed interactive loop for the AI agent and RTU terminal would benefit the artifact and allow for more complex workflows, such as user credential management, to be fulfilled. For known RTUs to the artifact, currently it receives a JSON file containing their IP addresses, but it could be enhanced by giving the AI agent a tool with which it can detect new devices within its LAN and receive their RTU basic information. Then they can be shown in the artifact user interface for the engineer to choose for certain configuration tasks. Additionally, a limitation of the current version of the artifact is that

it is only capable of using 10 commands of Hitachi's scripting interface, while the interface has over 80 commands and possible configuration tasks. Therefore, for full implementation of the artifact, all the commands would need to be tested and made examples from, which then can be used in fine-tuning or RAG for the final configuring agent to apply the knowledge for full workflows. Moreover, solving the limitations of multi-step autonomous workflows and project file navigation would enhance the artifact to a level of real project work adoption.

Further follow-up studies, which would develop the research on applying local AI agents to industrial processes, would be in the areas of AI agent integrations, sustainability studies, and machine learning studies on SLMs. For AI agent integration, two areas of AI agent autonomy with control over its own iterations of its tools, such as the terminal and its output, and implementing agentic AI where multiple SLMs are used, where one model develops the plan of actions for the executing model. Thus, allowing both to be specialized in their respective fields and areas. For further and more accurate estimations on the sustainability of SLMs and LLMs, their energy consumption metrics for agentic tasks should be studied to determine accurate numbers on their sustainability in terms of energy usage. In terms of machine learning, other SLM models should be evaluated for their performance on configuration workflow development, hence creating a more holistic picture of different SLM family capabilities. Another research aspect would be to continue fine-tuning for the qwen2.5-coder:1.5B model with a new, larger testing dataset to find out if the model can learn all the commands and their syntax, and where the limit of fine-tuning is reached by overfitting the model for the training data. Hence, further follow-up research shows three different directions with applied AI agents and integrating tools for them, sustainability metric comparisons between SLMs and LLMs, and fine-tuning comparisons of different SLM families.

5.2 Conclusion

5.2.1 Summary of research and answers to research questions

To feasibly implement AI agents into industrial workflows, they require a method of perception for their task and a tool to interact with it. Furthermore, to ensure safe and efficient performance, they require high-quality training data and examples of their preferred workflows. The research showed this feasibility through the file and backup management subprocesses, while showing limitations in credential management tasks.

Security is at the core of industrial processes; thus, the architectural design of the research's RTU configuring agent was created to be run locally in a separate network from malicious threats. The study confirmed that this security-first by design was feasible and resulted in the AI agent being capable of successfully running without sending any sensitive information to the cloud or storing sensitive information in any of its logs.

The performance of the AI agent showed fluctuations in the results compared to a human engineer's manual process. This was due to the similar workflow time in file management, while the backup workflow improved due to the design of the agent's application, allowing parallel configuration windows. Moreover, the user credential management performance was unsuccessful due to the agent's tool limitations. Hence, proving that the performance of an AI agent for the configuration processes is similar or better than the manual human work, if the limitations of tool usage and feedback are addressed. Thus, showing direction towards AI-assisted RTU configuration.

5.2.2 Significance, implications, and future research directions

The study brought together how industrial processes can be analyzed to determine their automation feasibility with local AI agents. In the single-case study for RTU configuration, the viability of using an AI agent in an industrial process was demonstrated. The study outlined boundaries for the agent with single-step and multi-step configurations, which can be applied to other processes underneath an automation lens. Therefore, the

research showed the significance that local AI agents will have in the future's Industry 5.0 and industrial automation.

Energy efficiency, local SLMs, and parallelization of configuration tasks were implications from the study's results. Proposed edge AI deployment, which is supported by research, proves to be feasible and more energy efficient than cloud AI in terms of sustainability. Data security and sovereignty were proven in the case study and supported by studies for local SLMs keeping organizational data securely within their premises. For RTU configuration, the possibility of parallelization opens new possibilities to configure a larger scale of RTUs. Thus, the study outlined implications for future energy-efficient edge AI deployment, data security benefits of local AI, and in the context of RTU configuration, the benefits of parallel RTU configuration.

Finally, the findings outline how energy efficiency and data security are benefits in utilizing lightweight and locally deployed AI agents. These agents provide a scalable path for operational environment automation in co-operation with humans, as industrial systems evolve towards more autonomous and data-rich workflows. This research, therefore, establishes local SLM-based agents as practical components for future industrial systems aligned with Industry 5.0 objectives.

References

Van der Aalst, W. M. P., Carmona, J., Weidlich, M., & Dongen, B. van. (Eds.). (2022). Process mining handbook. Springer. http://doi.org/10.1007/978-3-031-08848-3

Alotaibi, Y. (2020). A comprehensive analysis of business process modelling standards, techniques, and languages. *IJCSNS International Journal of Computer Science and Network Security*, 20(9), 233–250. https://doi.org/10.22937/IJCSNS.2020.20.09.29

Alzu'bi, S., Kanan, T., Elbes, M., Kanaan, G., & Trrad, I. (2025). Energy-efficient edge deployment of generative AI models using federated learning. *Cluster computing*, 28(5), 315. https://doi.org/10.1007/s10586-025-05263-7

Bandi, A., Kongari, B., Naguru, R., Pasnoor, S., & Vilipala, S. V. (2025). The Rise of Agentic AI: A Review of Definitions, Frameworks, Architectures, Applications, Evaluation Metrics, and Challenges. *Future Internet*, 17, 404. https://doi.org/10.3390/fi17090404

Bhargava, A., Bhargava, D., Kumar, P. N., Sajja, G. S., & Ray, S. (2022). Industrial IoT and AI implementation in vehicular logistics and supply chain management for vehicle mediated transportation systems. *International journal of system assurance engineering and management*, 13(Suppl 1), 673-680. https://doi.org/10.1007/s13198-021-01581-2

Chinthamu, N., Ashish, Emmanuel, V. P., Mathiyalagan, P., Rashid, S. Z., & Jothi, E. (2025). Robotic process automation in business processes streamlining operations through automation technologies. *ITM Web of Conferences*, 76, 01005. https://doi.org/10.1051/itmconf/20257601005

Choi, D., Rbigui, H., & Cho, C. (2021). Candidate digital tasks selection methodology for automation with robotic process automation. *Sustainability*, 13(16), 8980. https://doi.org/10.3390/su13168980

Chow, M., & Ng, O. (2025). Beyond chatbots: Moving toward multistep modular AI agents in medical education. *JMIR Medical Education*, 11(1), Article e76661. https://doi.org/10.2196/76661

Choudhary, S., S, V., Bhavani, D. D., N, B., Tiwari, M., & S, S. (2025). Edge AI Deploying Artificial Intelligence Models on Edge Devices for Real-Time Analytics. *ITM Web of Conferences* (Vol. 76, Article 01009). EDP Sciences. https://doi.org/10.1051/itmconf/20257601009

Corradini, F., Fornari, F., Polini, A., Re, B., & Tiezzi, F. (2021). A formal approach for the analysis of BPMN collaboration models. *Journal of Systems and Software*, 180, 111007. https://doi.org/10.1016/j.jss.2021.111007

Dumas, M., La Rosa, M., Mendling, J., & Reijers, H. A. (2020). Fundamentals of business process management (2nd ed.).

De Silva, D., Mills, N., Moraliyage, H., Rathnayaka, P., Wishart, S., & Jennings, A. (2025). Responsible Artificial Intelligence Hyper-Automation with Generative AI Agents for Sustainable Cities of the Future. *Smart cities (Basel)*, 8(1), 34. https://doi.org/10.3390/smartcities8010034

Eloundou, T., Manning, S., Mishkin, P., & Rock, D. (2023). GPTs are GPTs: An early look at the labor market impact potential of large language models. *Quantitative Finance*. 10, 2303. https://doi.org/10.48550/arXiv.2303.10130

El-Gharib, N. M., & Amyot, D. (2023). Robotic process automation using process mining — A systematic literature review. *Data & Knowledge Engineering*, 148, 102229. https://doi.org/10.1016/j.datak.2023.102229

European Parliament, & Council of the European Union. (2016). Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation). Official Journal of the European Union, L119, 1–88. https://eur-lex.europa.eu/eli/reg/2016/679/oj

Fraga-Lamas, P., Lopes, S. I., & Fernández-Caramés, T. M. (2021). Green IoT and Edge AI as key technological enablers for a sustainable digital transition towards a smart circular economy: An Industry 5.0 use case. *Sensors*, 21(17), 5745. https://doi.org/10.3390/s21175745

Gichamba, A., Idris, T. K., Ebiyau, B., Nyberg, E., & Mitamura, T. (2024). ColBERT Retrieval and Ensemble Response Scoring for Language Model Question Answering. https://doi.org/10.48550/arXiv.2408.10808

Griewing, S., Lechner, F., Gremke, N., Lukac, S., Jannis, W., Wallwiener, M., Wagner, U., Hirsch, M., & Kuhn, S. (2024). Proof-of-concept study of a small language model chatbot for breast cancer decision support - a transparent, source-controlled, explainable and data-secure approach. *Journal of Cancer Research and Clinical Oncology*, 150, 451 . https://doi.org/10.1007/s00432-024-05964-3

Hui, B., Yang, J., Cui, Z., Yang, J., Liu, D., Zhang, L., Liu, T., Zhang, J., Yu, B., Lu, K., Dang, K., Fan, Y., Zhang, Y., Yang, A., Men, R., Huang, F., Zheng, B., Miao, Y., Quan, S., ... Lin, J. (2024). Qwen2.5-Coder Technical Report. arXiv. https://doi.org/10.48550/arXiv.2409.12186

International Organization for Standardization. (2013). Information technology—Business Process Model and Notation (ISO/IEC 19510:2013). https://www.iso.org/standard/62652.html

Jegham, N., Abdelatti, M., Elmoubarki, L., & Hendawi, A. (2025). How hungry is AI? Benchmarking energy, water, and carbon footprint of LLM inference. arXiv. https://doi.org/10.48550/arXiv.2505.09598

Klesel, M., & Wittmann, H. F. (2025). Retrieval-Augmented Generation (RAG). *Business and Information Systems Engineering*, 67(4), 551–561. https://doi.org/10.1007/s12599-025-00945-3

Leng, J., Zhu, X., Huang, Z., Li, X., Zheng, P., Zhou, X., Mourtzis, D., Wang, B., Qi, Q., Shao, H., Wan, J., Chen, X., Wang, L., & Liu, Q. (2024). Unlocking the power of industrial artificial intelligence towards Industry 5.0: Insights, pathways, and challenges. *Journal of Manufacturing Systems*, 73, 349–363. https://doi.org/10.1016/j.jmsy.2024.02.010

Meuser, T., Lovén, L., Bhuyan, M., Patil, S. G., Dustdar, S., Aral, A., Bayhan, S., Becker, C., de Lara, E., Ding, A. Y., Edinger, J., Gross, J., Mohan, N., Pimentel, A. D., Rivière, E., Schulzrinne, H., Simoens, P., Solmaz, G., & Welzl, M. (2024). Revisiting Edge AI:

Opportunities and Challenges. *IEEE Internet Computing*, 28, 49–59. https://doi.org/10.1109/mic.2024.3383758

Parasuraman, R., Sheridan, T. B., & Wickens, C. D. (2000). A model for types and levels of human interaction with automation. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, 30(3), 286–297. https://doi.org/10.1109/3468.844354

Patrício, L., Varela, L., Silveira, Z., Felgueiras, C., & Pereira, F. (2025). A framework for integrating robotic process automation with artificial intelligence applied to Industry 5.0. *Applied Sciences*, 15, 7402 . https://doi.org/10.3390/app15137402

Perez-Cerrolaza, J., Abella, J., Borg, M., Donzella, C., Cerquides, J., Cazorla, F. J., Englund, C., Tauber, M., Nikolakopoulos, G., & Flores, J. L. (2024). Artificial intelligence for safety-critical systems in industrial and transportation domains: A survey. *ACM Computing Surveys*, 56(7), Article 176. https://doi.org/10.1145/3626314

Reslan, M., Triebe, M. J., Venketesh, R., & Hartwell, A. J. (2025). Automation of value stream mapping: A case study on enhancing lean manufacturing tools through digital twins. *Procedia CIRP,* 134, 455–460. https://doi.org/10.1016/j.procir.2025.02.159

Richardson, L. S., Fidock, J., & Gunawan, I. (2025). Systematic literature review of levels of automation (autonomy) taxonomy: Critiques and recommendations. *International Journal of Human-Computer Interaction.* https://doi.org/10.1080/10447318.2025.2502978

Rosenberg, S. M. (2021). The digitalization of the 21st century supply chain. Routledge.

Samuels, A. (2025). Examining the integration of artificial intelligence in supply chain management from Industry 4.0 to 6.0: A systematic literature review. *Frontiers in artificial intelligence*, 7, 1477044. https://doi.org/10.3389/frai.2024.1477044

Sapkota, R., Roumeliotis, K. I., & Karkee, M. (2026). AI Agents vs. Agentic AI: A Conceptual taxonomy, applications and challenges. *Information Fusion*, 126, 103599. https://doi.org/10.1016/j.inffus.2025.103599

Schultz, A. L. (2017). Integrating lean and visual management in facilities management using design science and action research. *Built Environment Project and Asset Management*, 7(3), 300–312. https://doi.org/10.1108/BEPAM-05-2016-0020

Singh, R., & Gill, S. S. (2023). Edge AI: A survey. *Internet of Things and Cyber-Physical Systems*, 3, 71–92. https://doi.org/10.1016/j.iotcps.2023.02.004

Strutzenberger, L., Mangler, J., & Rinderle-Ma, S. (2024). Evaluating BPMN extensions for continuous processes based on use cases and expert interviews. *Business & Information Systems Engineering*, 66(3), 389–405. https://doi.org/10.1007/s12599-023-00850-7

Vulpe, M., Stancu, S., Elena, D., Pernici, A., & Constantin, I. (2022). Robotic Process Automation through Process Mining. *Manager (University of Bucharest, Faculty of Business & Administration)*, 36(2).

Wang, F., Zhang, Z., Zhang, X., Wu, Z., Mo, T., Lu, Q., Wang, W., Li, R., Xu, J., Tang, X., He, Q., Ma, Y., Huang, M., & Wang, S. (2025). A comprehensive survey of small language models in the era of large language models: Techniques, enhancements, applications, collaboration with LLMs, and trustworthiness. *ACM Transactions on Intelligent Systems and Technology*. https://doi.org/10.1145/3768165

Zhang, Q., Liu, Z., & Pan, S. (2025). The rise of small language models. *IEEE Intelligent Systems*, 40, 30–37. https://doi.org/10.1109/mis.2024.3517792

Zupan, M. (2025). Developing an Accounting Virtual Assistant Through Supervised Fine-Tuning (SFT) of a Small Language Model (SLM*). Intelligent Systems in Accounting, Finance and Management*, 32, e70011. https://doi.org/10.1002/isaf.70011

Appendices

Appendix 1. File management performance snippet

File upload and update time	Manual (s)	Base model (s)	Fine-tuned model (s)
RTU A	71.88	61.04	62.59
RTU B	69.39	65.37	63.44
RTU560	75.07	72.98	88.64
Total	216.34	199.39	214.67

Note: A portion of the full dataset is shown here; complete data is available upon request.

Appendix 2. Power consumption during AI agent usage

Idle power of GPU (W)	Power during model queries (W)	Query duration (s)
4.11	32.54	5.00
4.43	32.16	5.00
4.27	40.46	5.00
4.34	39.76	4.00
4.03	39.05	5.00

Note: Power measurements were instantaneous and measured every second.