AI-Driven ERP Systems

Integrating Large Language Models for Enhanced Customer Interaction and Operational Efficiency

Elias Niederwieser, Dietmar Siegele und Dominik T. Matt*

Large Language Models (LLM) offer significant potential for automating complex tasks across domains. This article presents a novel Enterprise Resource Planning (ERP) system that leverages LLM to fulfill customer requests by accessing the ERP database for real-time updates, modifications, and availability checks, enhancing interaction and efficiency. Using a graph-theoretic framework, the system supports stateful workflows with cycles, branching, and human-in-the-loop (HITL) interactions, allowing precise control over application flow. This approach redefines LLM deployment in ERP applications, providing enhanced automation and responsiveness in customer service.

Introduction

ERP systems streamline core business processes across various functions [1, 2]. Traditional ERP systems are often costly and complex, posing challenges for small to medium-sized enterprises (SMEs) [3, 4]. Advances in artificial intelligence, particularly LLM like GPT-4 and Llama 3, transform ERP systems by providing flexible, scalable solutions for SMEs. By enabling natural language interactions and automating tasks, LLM helps streamline ERP functions with reduced cost and complexity [5–7]. Integrated LLM can improve efficiency and customer satisfaction by providing real-time responses and seamless database updates [8, 9].

Incorporating AI into ERP workflows requires careful control to ensure data security, privacy, and reliable oversight [10, 11]. This study proposes a novel AI-driven ERP system that leverages LLM using a graph-theoretical approach with HITL mechanisms to maintain essential control while benefiting from automation [12].

The study aims to:

- develop an AI-driven ERP system that uses natural language for efficient customer interactions,
- automate core functions like inventory management and order processing to reduce manual workload, as well as
- ensure security and compliance with HITL mechanisms for sensitive transactions.

State of the Art

Significant advancements have been made in integrating artificial intelligence into ERP systems, with numerous studies exploring machine learning and natural language processing (NLP) for process automation and enhanced customer interactions. Research has demonstrated that traditional ERP systems can be enhanced using AI to automate repetitive tasks, optimize resource allocation, and improve decision-making through data analysis [13, 14]. However, these systems often lack the flexibility to handle natural language queries and require extensive customization for specific workflows.

Several studies have incorporated NLP to interpret customer queries and automate simple workflows [15]. Nonetheless, these systems rely on predefined templates and fail to adapt dynamically to complex or evolving scenarios. HITL methods have been explored in decision-critical workflows to maintain oversight and security [16]. While effective in ensuring control, these approaches often introduce inefficiencies due to the reliance on human intervention at multiple stages.

Recent developments in graph-theoretical approaches have facilitated the creation of more adaptive workflows in ERP systems [17]. Yet, these models frequently lack integration with stateful AI agents capable of leveraging LLM for contextual reasoning.

* Corresponding Author
Elias Niederwieser, M. Sc., B. Sc.; Fraunhofer Italia Research; Via A. Volta 13 A, 39100 Bolzano, Italy; Phone: +39 393 9649282, E-Mail: elias.niederwieser@fraunhofer.it

Co-authors
Dietmar Siegele, M. Sc.; Fraunhofer Italia Research
Prof. Dr.-Ing. Dominik T. Matt; Fraunhofer Italia Research and Free University of Bozen-Bolzano

Note
This article is peer reviewed by the members of the ZWF Special Issue Advisory Board.

Research Gaps Addressed by Our Solution

Despite these advancements, the following limitations persist in existing systems, which our solution addresses:

- *Dynamic Natural Language Interaction*
  Unlike template-driven NLP implementations, our system employs LLM for real-time interpretation of complex, natural language customer requests, improving usability and reducing manual inputs.
- *Adaptive and Stateful Workflow Management*
  Our graph-theoretical framework with state persistence enables the system to dynamically adjust workflows, handle branching and iterative processes, and incorporate human input seamlessly when necessary.
- *Human-in-the-loop with Minimal Overhead*
  By combining HITL mechanisms with AI-driven checkpoints and validation, our approach ensures compliance and security without excessive manual intervention, striking an optimal balance between automation and oversight.
- *Scalability and Simplicity for SME*
  Many state-of-the-art solutions are resource-intensive, making them unsuitable for SMEs. Our streamlined architecture, leveraging only a minimal set of tools, provides cost-effective scalability.

The proposed AI-driven ERP system establishes a novel paradigm for integrating LLM into enterprise workflows, addressing these gaps to ensure adaptability, efficiency, and enhanced customer satisfaction.

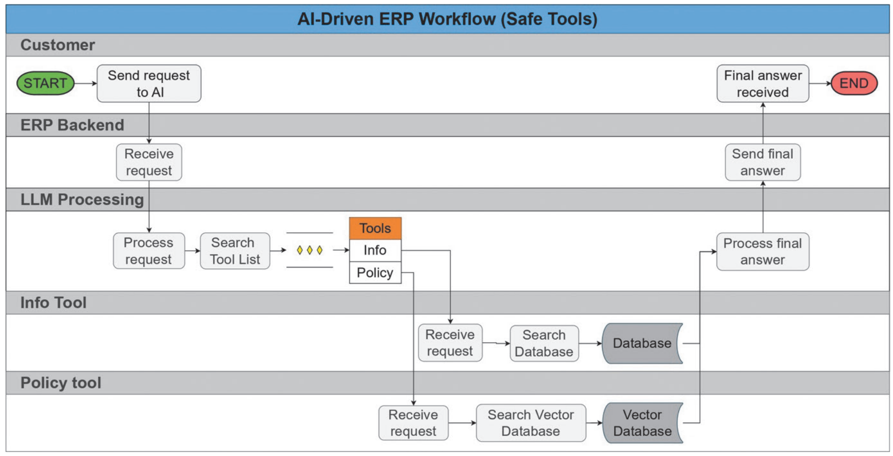

Figure 1. The Safe Tools Workflow manages information-only requests, where the AI agent retrieves data without modifying records. Info and Policy Tools ensure compliance and provide quick responses without human intervention

Designing the AI-Driven ERP System Architecture

This article presents a streamlined AI-driven ERP system with a core architecture built around two primary tables: the Product Table and the Ordering Table. The Product Table holds product details, stock levels, incoming deliveries, and expected arrival times, supporting accurate inventory management. The Ordering Table records transaction details for order processing and record-keeping.

Client requests submitted via an Application Programming Interface (API) receive a unique identifier and are routed to an AI agent powered by an LLM [18]. The agent interprets natural language requests to understand user intent and selects appropriate tools to address them. While a single tool often suffices, complex requests may involve several tools.

Incorporating Safe and Sensitive Tools

The system distinguishes between Safe and Sensitive Tools for security and operational integrity:

- *Safe Tools*
  For data retrieval, the agent uses the Info Tool to query the Product Table and provide relevant details without modifying data, enabling accurate customer responses. The Policy Tool employs a Retrieval-Augmented Generation (RAG) technique to access a policy document in a vector database, covering access control, data privacy, delivery protocols, inventory management, and return policies [19]. This ensures the AI agent's actions comply with company policies and regulations.
- *Sensitive Tools*
  The Ordering Tool updates both the Ordering Table and the Product Table, recording transactions and adjusting stock levels. As a HITL tool, the Order Placement Tool requires explicit customer confirmation before proceeding, ensuring acknowledgment of transaction details. The Escalation Tool routes cases needing further scrutiny to a human support representative.

Workflow Management and Compliance Assurance

The AI agent orchestrates the workflows depicted in Figure 1 and Figure 2, managing request contexts and interactions across components. These workflows demonstrate how client requests are handled securely and efficiently by integrating Safe and Sensitive Tools, along with HITL and escalation protocols. By leveraging these mechanisms, the system ensures compliance with organizational policies and regulatory standards while maintaining operational excellence.

The system employs a specialized AI agent architecture to effectively implement and manage these workflows. In the next section, we delve into the structure and functionality of the Single-Shot ReAct AI Agent, which is crucial for dynamic workflow management and the integration of human oversight within the ERP system [20].

Implementing the Single-Shot ReAct AI Agent

The Single-Shot ReAct AI Agent is a key component of the proposed ERP system. It is designed to efficiently handle user requests through reactive reasoning and single-step decision-making. To implement the agent, we utilize the open-source Python LangGraph library, a graph-based workflow framework inspired by Pregel and Apache Beam, to

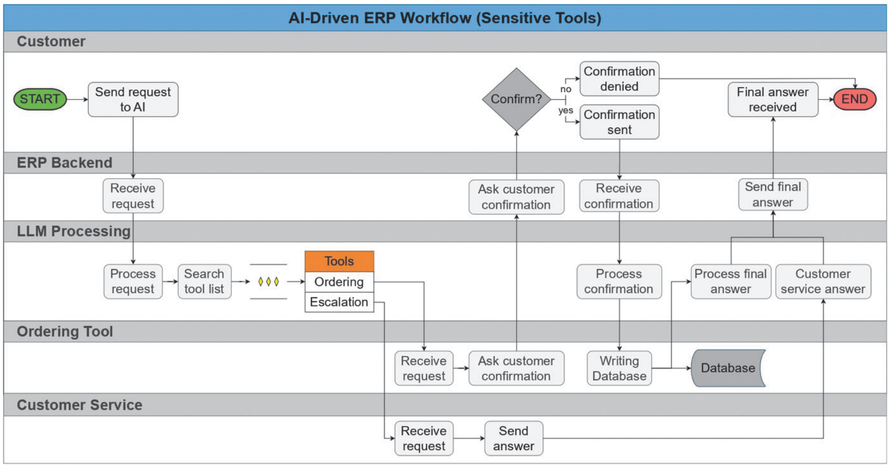

Figure 2. The Sensitive Tools Workflow handles critical ERP tasks requiring data updates, like order placements. The AI agent processes requests, but customer confirmation is required before changes. Complex cases are escalated to human support, ensuring secure handling of sensitive operations

manage complex AI-driven workflows with dynamic adaptability and reliable persistence [21–23].

Structure and Functional Overview
A graph model is at the core of the Single-Shot ReAct agent, where each node represents a distinct action, decision, or tool, and edges define the logical flow between these nodes. This setup allows the agent to navigate diverse decision paths based on real-time context. Each node evaluates the system's current state before progressing, enabling the agent to adaptively backtrack, iterate, or re-evaluate steps as task requirements evolve. Unlike static workflows, which follow a set sequence, this dynamic structure supports flexible and responsive task management.

Key Components of the AI Agent
Below is a concise overview of the fundamental components that empower the AI agent's dynamic, context-aware decision-making capabilities.

- *State*
  Acting as the agent's memory, the state continuously updates with conversation history, task-specific data, and interim results. This persistent state enables the agent to retain context across interactions, handle interruptions smoothly, and incorporate human adjustments as needed.
- *Nodes and Transition Functions*
  Each graph node represents a specific function or operation within the workflow, ranging from standard language model interactions to external tool calls. Transition functions govern how the state updates at each step, allowing nodes to modify or expand the state based on real-time conditions, enhancing response adaptability and accuracy.
- *Edges and Condition Functions*
  The edges between nodes define decision pathways. Conditional edges, in particular, allow the agent to assess specific criteria before advancing, enabling intelligent branching. This feature is critical for workflows with conditional branches or for those requiring loops that revisit nodes to gather more information, creating an intelligent and context-aware decision-making process.

Persistence and Human-in-the-Loop Control Mechanisms
LangGraph's persistence layer ensures comprehensive workflow monitoring and controls through checkpoints, which save snapshots of the agent's state and workflow metadata at each step. Each workflow is assigned a unique identifier, allowing developers to retrieve checkpoints, examine the exact state, view pending tasks, and access metadata related to the last executed node.

For HITL control, developers can set breakpoints or conditions to pause the workflow for review or user approval. If an action requires human validation, the workflow halts, enabling users to review the state, adjust, or approve the task before it continues. These features provide a controlled and adaptable framework, balancing AI autonomy with human oversight essential for managing complex workflows.

Demonstrating the System with Tested Workflows

This section showcases workflows highlighting our AI-driven ERP system's practical application and effectiveness, tailored to specific customer and operational needs. The Single-Shot ReAct AI agent efficiently manages diverse requests and complex scenarios within a scalable, controlled environment.

Prerequisites and Setup for Workflow Execution
To achieve optimal performance and stability, the following configurations were implemented:

- *Database*
  For real-time accuracy, a relational SQL database manages inventory, transactions, and customer interactions with two primary tables: Product and Ordering. The database contains 500 unique coffee bean products for testing, ensuring robust handling of customer inquiries and orders.
- *AI Agent Orchestration*
  LangGraph v0.2 provides a flexible, stateful structure with checkpoints for tracking, review, and HITL interventions. It utilizes ChatGPT-4o as the LLM, enabling adaptive, graph-driven workflows for dynamic decision-making.
- *Tool Configuration*
  Python functions with SQL code allow the AI agent to perform data retrieval and transactional tasks directly within workflows. Secure tools facilitate distinct data retrieval and transactions.

These workflows exemplify best practices in AI-driven ERP setups, balancing automation with essential human oversight. Each flowchart below illustrates the structured approach for different scenar-

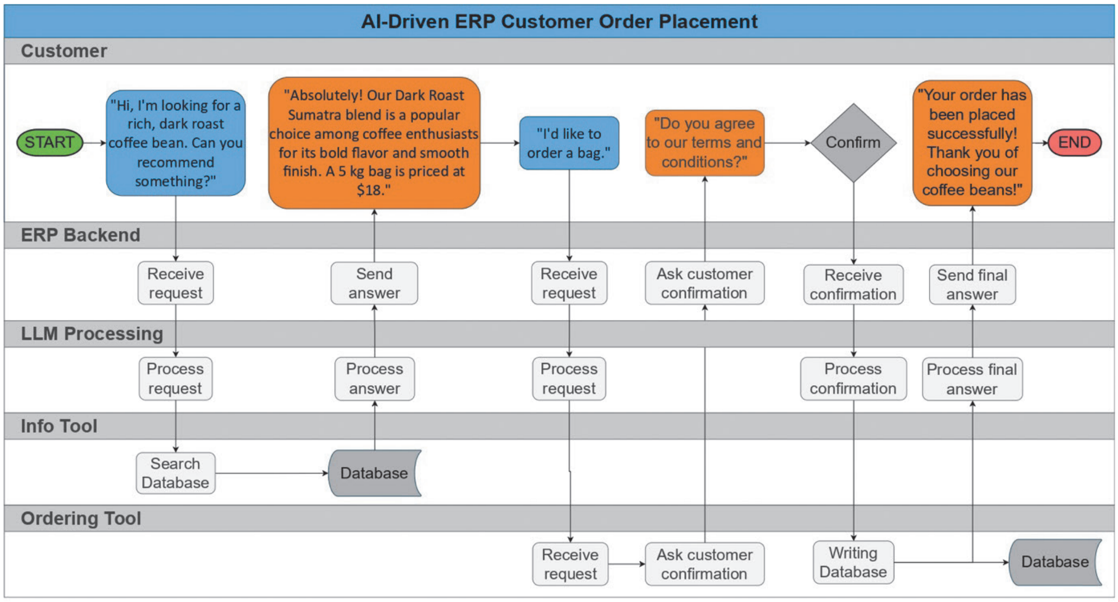

Figure 3. This diagram illustrates a customer order placement process. The customer starts by requesting a coffee recommendation. The LLM processing layer interprets customer inputs and manages requests, with the Info Tool querying the database for information. If the customer decides to order, the Ordering Tool requests confirmation and updates the database. The final step is a confirmation message to the customer, completing the order process

ios, highlighting the system's adaptability across various operational contexts.

Case Studies and Practical Examples

To evaluate the effectiveness of the AI-integrated ERP system, we present two case studies demonstrating its capabilities in real-world scenarios. These examples illustrate how the system handles customer interactions, processes orders, and incorporates human-in-the-loop mechanisms to ensure security and customer satisfaction.

Workflow for Customer Order Placement
This workflow (see Figure 3) outlines the process of assisting a customer in selecting and purchasing coffee beans. The flowchart highlights each step, from the initial product recommendation to order confirmation, emphasizing seamless interaction and customer consent. This process involves the Info Tool for retrieving product details and the Ordering Tool for processing the transaction. As depicted in Figure 3, the AI agent ensures smooth communication and verifies customer agreement before finalizing the order, demonstrating the system's capacity for efficient and customer-friendly service.

Workflow for Customer Order Return
This workflow (see Figure 4) demonstrates the system's capability to manage customer return inquiries autonomously while incorporating mechanisms for human intervention when necessary. In this scenario, the customer initiates a return request due to dissatisfaction with a product. The AI agent first responds by retrieving policy information through the Policy Tool to inform customers of their options. If further assistance or verification is required, the Escalation Tool activates, forwarding the case to a human customer service representative, as shown in Figure 4. This escalation ensures that while routine inquiries are handled autonomously, complex or sensitive cases receive human attention to uphold service quality and customer satisfaction.

These tested workflows and case studies underscore the AI ERP system's adaptability and effectiveness across various operational contexts. It balances automation with essential human oversight to optimize performance and enhance customer satisfaction in industrial and research-driven environments.

Conclusion and Future Directions

Integrating LLM into ERP systems marks a significant leap in automating customer interactions and enhancing operational efficiency. This approach offers SMEs a scalable, cost-effective way to manage resources and improve customer satisfaction. The tested workflows – Customer Order Placement and Customer Order Return – demonstrate the system's ability to handle routine tasks effectively while maintaining customer satisfaction through a balance of automation and HITL mechanisms.

However, relying on only four tools may limit the system's capacity to manage more complex scenarios. Future developments may focus *subgraphs on multi-agent systems* and reflection mechanisms, where specialized agents, managed by an LLM supervisor, can handle complex workflows and adapt to evolving contexts [24, 25]. Incorporating reflection mechanisms enables the LLM to learn from past actions, leading to more accurate and adaptive responses [26].

Adding *multimodal* capabilities would further enhance the system by allowing the LLM to process various data types, such as images, making the ERP more versatile [27, 28]. Future research should involve real-world testing with SME to assess practical applicability and explore technological improvements like advanced LLM with reflective and multimodal abilities and optimized frameworks supporting multi-agent orchestration.

Addressing these areas can evolve the AI-driven ERP system to offer greater automation, efficiency, and responsiveness, providing SMEs with a powerful tool for effective resource management and cus-

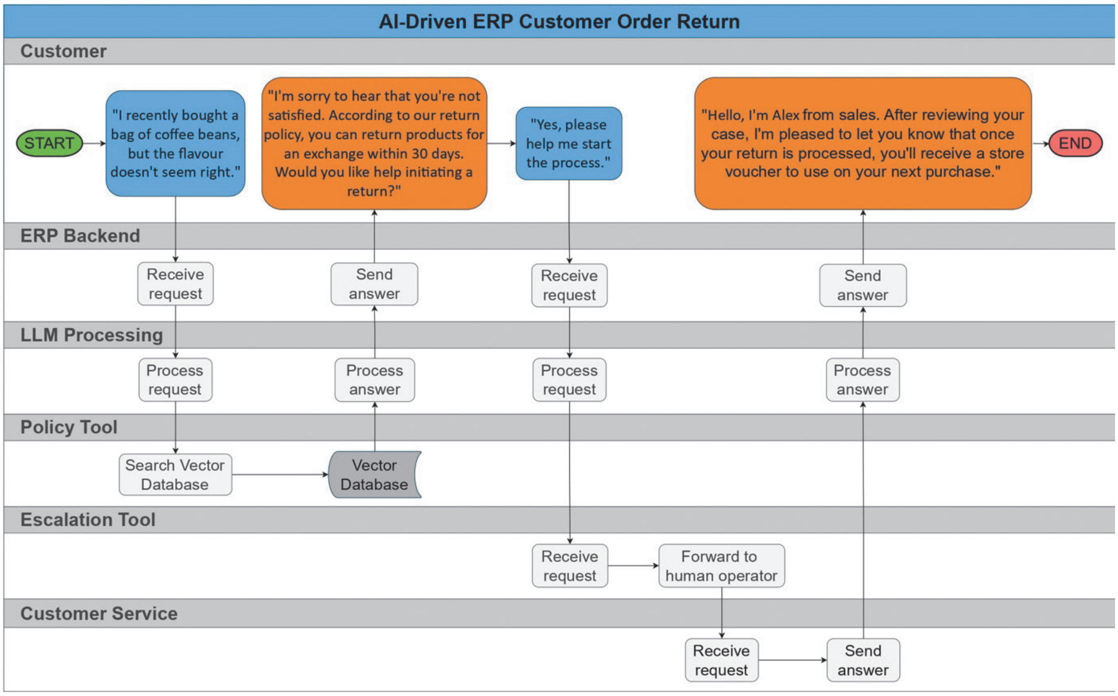

Figure 4. This diagram shows a customer order return process. It begins with a customer complaint about a product. The LLM processing layer interprets and manages customer requests, while the Policy Tool references a vector database to ensure responses align with return policies. If needed, the Escalation Tool forwards the case to Customer Service for further assistance. The process concludes with a personalized message from sales, notifying the customer of a voucher for future purchases

tomer interactions in an increasingly digital marketplace.

Literature

1. Rosemann, K. M.; Gable, G. G.: What is ERP? Information Systems Frontiers 2 (2000) 2, pp. 141–162 DOI:10.1023/A:1026543906354
2. Jacobs, F. R.; Weston, F. C. Jr.: Enterprise Resource Planning (ERP) – A Brief History. Journal of Operations Management 25 (2007) 2, pp. 357–363 DOI:10.1016/j.jom.2006.11.005
3. Buonanno, G.; Faverio, P.; Pigni, F. et al.: Factors Affecting ERP System Adoption: A Comparative Analysis Between SMEs and Large Companies. Journal of Enterprise Information Management 18 (2005) 4, pp. 384–426 DOI:10.1108/17410390510609572
4. Haddara, M.; Zach, O.: ERP Systems in SMEs: A Literature Review. Procedia Technology 5 (2012), pp. 99–104 DOI:10.1109/HICSS.2011.191
5. OpenAI: GPT-4 Technical Report. 2023 DOI:10.48550/arXiv.2303.08774
6. Touvron, H.; Martin, L.; Stone, K. et al.: Llama 2: Open Foundation and Fine-Tuned Chat Models. 2023 DOI:10.48550/arXiv.2307.09288
7. Zdravković, M.; Panetto, H.: Artificial Intelligence-enabled Enterprise Information Systems. Enterprise Information Systems 16 (2021) 5 DOI:10.1080/17517575.2021.1973570
8. Bawa, S. S.: Automate Enterprise Resource Planning with Bots. International Journal of Computer Trends and Technology 72 (2024) 1, pp. 83–91 DOI:10.14445/22312803/IJCTT-V72I1P114
9. Ram, J.; Corkindale, D.; Wu, M.-L.: Examining the Role of System Quality in ERP Projects. Industrial Management & Data Systems 113 (2013) 3 DOI:10.1108/02635571311312659
10. Brundage, M.; Avin, S.; Clark, J. et al.: The Malicious Use of Artificial Intelligence: Forecasting, Prevention, and Mitigation. 2018 DOI:10.48550/arXiv.1802.07228
11. Goodman, B.; Flaxman, S.: European Union Regulations on Algorithmic Decision-making and a "Right to Explanation". AI Magazine 38 (2017) 3, pp. 50–57 DOI:10.1609/aimag.v38i3.2741
12. Shneiderman, B.: Human-centered Artificial Intelligence: Reliable, Safe & Trustworthy. International Journal of Human-Computer Interaction 36 (2020) 6, pp. 495–504 DOI:10.1080/10447318.2020.1741118
13. Sisyukov, A. N.; Bondarev, V. K.; Yulmetova, O.: ERP Data Analysis and Visualization in High-Performance Computing Environment. 2020 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus) (2020), pp. 509–512 DOI:10.1109/EICon-Rus49466.2020.9038949
14. Dong, A.: ERP and Artificial Intelligence based Smart Financial Information System Data Analysis Framework. 2021 6th International Conference on Inventive Computation Technologies (ICICT) (2021), pp. 845–848 DOI:10.1109/ICICT50816.2021.9358659
15. Adebiyi, A.; Apeh, F.; Olaniyan, J.; Adebiyi, M. O.; Olaniyan, D.; Oluwasegunfunmi, I.-B.; Akindeji, K.: Automating Customer Service Using Natural Language Processing. 2024 International Conference on Science, Engineering and Business for Driving Sustainable Development Goals (SEB4SDG) (2024), pp. 1–8 DOI:10.1109/SEB4SDG60871.2024.10630201
16. Zhao, T.; Taherisadr, M.; Elmalaki, S.: FAIRO: Fairness-aware Adaptation in Sequential-Decision Making for Human-in-the-Loop Systems. ArXiv (2023) DOI:10.48550/arXiv.2307.05857
17. Fu, Z.; Xian, Y.; Zhu, Y.; Xu, S.; Li, Z.; de Melo, G.; Zhang, Y.: HOOPS: Human-in-the-Loop Graph Reasoning for Conversational Recommendation. In: Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval (2021) DOI:10.1145/3404835.3463247
18. Brown, T. B.; Mann, B.; Ryder, N. et al.: Language Models are Few-shot Learners. 2020 DOI:10.48550/arXiv.2005.14165
19. Lewis, P.; Perez, E.; Piktus, A. et al.: Retrieval-augmented Generation for Knowledge-intensive NLP Tasks. Advances in Neural Information Processing Systems 33 (2020), pp. 9459–9474 DOI:10.48550/arXiv.2005.11401
20. Yao, S.; Zhao, P.; Yu, H. et al.: Synergizing Reasoning and Acting in Language Models.

2022
DOI:10.48550/arXiv.2210.03629
21. Malewicz, G.; Austern, M.H.; Bik, A.J.C. et al.: Pregel: A System for Large-scale Graph Processing. In: Proceedings of the 2010 ACM SIGMOD International Conference on Management of Data. ACM 2010, pp. 135–146 DOI:10.1145/1807167.1807184
22. Chambers, C.; Raniwala, A.; Perry, F. et al.: FlumeJava: Easy, Efficient Data-parallel Pipelines. In: Proceedings of the 31st ACM SIGPLAN Conference on Programming Language Design and Implementation. ACM 2010, pp. 363–375 DOI:10.1145/1809028.1806638
23. Akidau, T.; Bradshaw, R.; Chambers, C. et al.: The Dataflow Model: A Practical Approach to Balancing Correctness, Latency, and Cost in Massive-scale, Unbounded, Out-of-order Data Processing. In: Proceedings of the VLDB Endowment 8 (2015) 12, pp. 1792–1803 DOI:10.14778/2824032.2824076
24. Kamar, E.: Directions in Hybrid Intelligence: Complementing AI Systems with Human Intelligence. In: Proceedings of the 25th International Joint Conference on Artificial Intelligence. AAAI 2016, pp. 4070–4073
25. Wooldridge, M.: An Introduction to MultiAgent Systems. 2nd Edition, Wiley, Chichester 2020
26. Shinn, N.; Cassano, F.; Berman, E. et al.: Reflexion: Language Agents with Verbal Reinforcement Learning. 2023 DOI:10.48550/arXiv.2303.11366
27. Radford, A.; Kim, J.W.; Hallacy, C. et al.: Learning Transferable Visual Models from Natural Language Supervision. In: Proceedings of the 38th International Conference on Machine Learning. PMLR, 2021, pp. 8748–8763 DOI:10.48550/arXiv.2103.00020
28. Alayrac, J.-B.; Donahue, J.; Luc, P. et al.: Flamingo: A Visual Language Model for Few-shot Learning. 2022 DOI:10.48550/arXiv.2204.14198

Authors

Elias Niederwieser, M.Sc., born in 1993, holds a Master's degree with distinction in Mathematical and Theoretical Physics from the University of Vienna. After completing his studies, he joined the Mathematical Physics research team at the University of Vienna, specializing in quantum gravity. In this role, he applied advanced mathematical computations to develop and analyze theoretical models addressing the fundamental forces and the structure of spacetime. Since 2022, he has been with Fraunhofer Italia in Bolzano, Italy, where his research now spans a diverse range of topics within artificial intelligence and machine learning. At Fraunhofer Italia, he leverages his strong foundation in theoretical physics and computational techniques to design and implement innovative AI and ML solutions, thereby supporting the organization's scientific and technological initiatives.

Dr. Dietmar Siegele, born in 1987, earned a Ph.D. in Engineering Sciences from the University of Innsbruck, where he also obtained Master's degrees in Domotronics and Structural and Civil Engineering. From 2014 to 2019, he served as a scientific research assistant at the University of Innsbruck, focusing on the development of heat pumps and the enhancement of energy efficiency in building systems. In 2019, Dr. Siegele joined Fraunhofer Italia in Bolzano, Italy, where he initially concentrated on augmented and virtual reality applications for collaborative robotics, as well as AI-driven solutions to improve energy efficiency. Since 2022, he has been the Head of Process Engineering in Construction at Fraunhofer Italia, overseeing projects related to AI, process optimization in construction, sustainable building practices, and automation, thereby advancing efficient and sustainable construction technologies.

Prof. Dr.-Ing. Dominik Matt holds the Chair of Production Systems and Technologies (IIND-04/A) and leads the Industrial Engineering & Automation research area in the Faculty of Engineering at the Free University of Bozen-Bolzano. He is also the head of the Fraunhofer Italia Research Institute in Bozen-Bolzano. With over 350 scientific and technical publications, he is recognized among the most cited professors globally, ranking within the top 2 % of scientists in his field (elsevier.digitalcommonsdata.com, 08/2024). Prof. Matt is an active member of numerous scientific organizations and committees, and since 2020, he has been a member of the prestigious German National Academy of Science and Engineering.

Abstract

KI-gesteuerte ERP-Systeme. Große Sprachmodelle (LLM) bieten erhebliches Potenzial zur Automatisierung komplexer Aufgaben in verschiedensten Anwendungsbereichen. Dieser Artikel präsentiert ein neuartiges Enterprise-Resource-Planning (ERP)-System, das LLM einsetzt, um Kundenanfragen zu erfüllen, indem es auf die ERP-Datenbank zugreift und in Echtzeit Aktualisierungen, Änderungen sowie Verfügbarkeitsprüfungen durchführt – was Interaktion und Effizienz steigert. Mithilfe eines graphentheoretischen Rahmens unterstützt das System zustandsbehaftete Workflows mit Zyklen, Verzweigungen und Human-in-the-Loop (HITL)-Interaktionen, wodurch eine präzise Steuerung des Anwendungsflusses ermöglicht wird. Dieser Ansatz definiert den Einsatz von LLM in ERP-Anwendungen neu und bietet eine verbesserte Automatisierung sowie eine erhöhte Reaktionsfähigkeit im Kundenservice.

Keywords

AI-Driven ERP-Systems, Large Language Model Integration, Human-in-the-Loop (HITL) Mechanisms, Graph-Theoretic Workflow Framework, Natural Language Processing (NLP), Workflow Automation

Schlüsselwörter

KI-gesteuerte ERP-Systeme, Integration großer Sprachmodelle, Human-in-the-Loop (HITL)-Mechanismen, Graphentheoretisches Workflow-Framework, Natürliche Sprachverarbeitung (NLP), Workflow-Automatisierung

Bibliography

DOI:10.1515/zwf-2025-0007
ZWF 120 (2025) Special Issue; page 112 – 117

ISSN 0947-0085 · e-ISSN 2511-0896